Statistical Analysis for Network Data using Matrix Variate Models and Latent Space Models

by

Xuefei Zhang

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Statistics)
in The University of Michigan
2020

Doctoral Committee:

Professor Ji Zhu, Chair
Professor Elizaveta Levina
Assistant Professor Gongjun Xu
Associate Professor Xiang Zhou

Xuefei Zhang
xfzhang@umich.edu
ORCID iD: 0000-0003-3266-0364

TABLE OF CONTENTS

LIST OF FIGURES v

LIST OF TABLES vii

LIST OF APPENDICES viii

ABSTRACT ix

CHAPTER

I. Introduction 1

II. Prediction for Network-linked Data using Matrix Variate Models 5

2.1 Introduction 5
2.1.1 Main Contributions 6
2.1.2 Related Work 8
2.1.3 Notations and Organization 9
2.2 Proposed Methods 10
2.2.1 Matrix Variate Model with Kronecker Sum Covariance 10
2.2.2 Parameter Estimation 14
2.2.3 Regression on Network-linked Data 18
2.2.4 Classification Setting 22
2.3 Theoretical Properties 23
2.4 Numerical Studies 25
2.4.1 Simulation Studies: Regression 26
2.4.2 Simulation Studies: Classification 30
2.4.3 NASA Central America Grid Data Example 32
2.5 Discussion and Future Work 34

III. Joint Latent Space Models for Network Data with High-dimensional Node Variables 36

3.1 Introduction . . . 36
3.2 Proposed Method . . . 40
3.2.1 Joint Latent Space Model . . . 40
3.2.2 Estimation . . . 44
3.3 Theoretical Results . . . 46
3.4 Simulation Studies . . . 51
3.4.1 Effect of the Dimension of Node Variables . . . 51
3.4.2 Effect of Network Density . . . 54
3.4.3 Community Membership Estimation . . . 55
3.4.4 Node Variable Missing Value Imputation . . . 58
3.5 Real Data Example . . . 58
3.6 Conclusion and Discussion . . . 62

IV. A Flexible Latent Space Model for Multilayer Networks . . . 64

4.1 Introduction . . . 64
4.2 Related Work . . . 66
4.3 Proposed Model . . . 68
4.3.1 Latent Space Model for Multilayer Networks . . . 68
4.3.2 Parameter Estimation . . . 70
4.4 Theoretical Results . . . 72
4.5 Simulation Studies . . . 74
4.6 Real Data Applications . . . 78
4.6.1 Lazega Lawyers Data . . . 78
4.6.2 Karnataka Data . . . 81
4.7 Conclusion and Discussion . . . 82

APPENDICES . . . 83
A.1 Proof of Theorems . . . 84
A.1.1 Proof of Theorem II.7 . . . 84
A.1.2 Proof of Theorem II.11 . . . 89
A.2 EM algorithm for Parameter Estimation . . . 94
A.2.1 Estimation for Zero-mean Matrix Variate Model . . . 94
A.2.2 Extension to Classification Setting . . . 98
A.2.3 Efficient Calculation . . . 100
A.3 Predicting Class Labels by Variational Methods . . . 103
B.1 Proof of Theorem III.5 . . . 105
B.2 Proof of Theorem III.9 and Proposition III.10 . . . 109
B.2.1 Proof of Proposition III.10 . . . 110
B.2.2 Proof of Theorem III.9 . . . 117
B.2.3 Lemmas on Initial Estimates with Good Properties 119
C.1 Proof of Theorem IV.3 . . . 122
C.1.1 Lemmas for Theorem IV.3 . . . 126
C.2 Proofs of Theorem IV.4 and Corollary IV.6 . . . 127

C.3 Proof of Proposition IV.1 128
C.4 Additional Simulation Results 130

BIBLIOGRAPHY 133

LIST OF FIGURES

Figure

2.1 In-sample prediction, $n = 200, p = 2$. Left: MSE vs $\log(\tau^2)$, $\gamma = 1$; right: MSE vs $\log(\gamma)$, $\tau^2 = 5$. 28
2.2 Semi-supervised learning, $n = 200, p = 2$. MSE vs the proportion of observed responses, $\tau^2 = 2$, $\gamma = 1$. 29
2.3 In-sample prediction, $n = 50, p = 19$. Left: $\Sigma \sim$ AR(1); right: $\Sigma_{xx} = I_p$. 30
2.4 Test accuracy vs μ_0. Left: $n = 100$, $p = 3$; right: $n = 50$, $p = 20$. . . 32
2.5 Visualization of the grid 33
3.1 Graphical representation of the joint latent space model 41
3.2 Estimation of Z versus Dimension of Y 53
3.3 Optimal λ versus Dimension of Y 54
3.4 Estimation of Z versus Network Density 56
3.5 Estimation of Z for Community Membership 57
3.6 Missing Value Imputation Results 59
3.7 Example of an ego-network: (a) Circle 1 network; (b) node variable matrix; (c) number of variables vs mean of the variable. 60
4.1 (a) and (b): Estimation error of parameters when $n = 400$, $R = 100$ and $k = 2$. Each light blue curve corresponds to one replication; the black curve corresponds to the average of all replications. The red dashed line in (b) corresponds to the line whose intercept and slope equal to the average of fitted intercepts and slopes respectively. (c): Histogram of all fitted slopes. 76
4.2 Average running time per iteration in seconds with one-standard-deviation error bars. Left: $n = 200$, and the R-square of a linear model is 0.998; Right: $R = 200$, and the R-square of a quadratic model is 0.998. 78
4.3 Visualization of the Lazega lawyer data. Nodes in different layers exhibit different connecting patterns. For example, the two red nodes are isolated in the friendship network but have several links in other layers, while the blue node is not connected to other nodes in the co-worker network but is well-connected in the friendship and advice networks. 79

4.4 Upper row: estimated U based on single networks. Lower row: jointly estimated U based on multilayer networks using different methods. Color represents the lawyer's office. 79
4.5 Link Prediction: AuROC on the test sets for 12 layers of the Karnataka Data. 80
C.1 (a) and (b): Estimation error of parameters when $n = 200$, $R = 50$ and $k = 2$. Each light blue curve corresponds to one replication; the black curve corresponds to the average of all replications. The red dashed line corresponds to the line whose intercept and slope equal to the average fitted intercepts and slopes. (c): Histogram of all fitted slopes. 131
C.2 (a) and (b): Estimation error of parameters when $n = 400$, $R = 100$ and $k = 4$. Each light blue curve corresponds to one replication; the black curve corresponds to the average of all replications. The red dashed line corresponds to the line whose intercept and slope equal to the average fitted intercepts and slopes. (c): Histogram of all fitted slopes. 132

LIST OF TABLES

Table

2.1 R^2 for predicting the temperature 34

3.1 Node variable missing value imputation results for Facebook data. Upper: AUC obtained by $\widehat{Z}_{net}$; Lower: AUC obtained by $\widehat{Z}_{joint}$. . . 61

LIST OF APPENDICES

Appendix

A. Appendix of Chapter 2 84

B. Appendix of Chapter 3 105

C. Appendix of Chapter 4 122

ABSTRACT

Network data capture the connectivity relationship among individuals and are ubiquitous in many scientific and engineering fields. This thesis focuses on developing statistical learning methodologies and novel statistical models for network data appearing in modern big data era.

Classical supervised learning methods usually assume the training data points are independent samples. However, when individuals are connected by a network and interact in complex ways, the classical independence assumption may not hold. In such a scenario, incorporating the network information in modeling is expected to improve the prediction performance, as it provides additional information about relationships among individuals. We first focus on predicting a continuous response variable using both covariates and network information. Specifically, we propose a matrix variate model that allows two-way dependence among data points and among variables, to model the distribution of variables associated with nodes in a network. Under such a model, the derived distribution of each response depends on covariates of all the data points in the network in a principled way. We develop efficient algorithms for parameter estimation and also show consistency of the estimators under mild conditions. Further, we extend the proposed framework to handle the classification problem.

The dimension of variables associated with nodes can be high in many modern data applications and such node variables usually provide important information for understanding network structure. In the second project, we consider the problem of modeling network data with node variables. The classical network latent space model

assumes that the edge formation in a network depends on nodal latent variables as well as the observed node variables, however, it has several limitations to handle high-dimensional node variables. We propose an alternative model, named joint latent space model, where we assume that the latent variables not only explain the network structure, but also are informative for the multivariate node variables. We establish theoretical properties of the estimators and provide insights on how incorporating high-dimensional node variables could improve the estimation accuracy of the latent positions. We demonstrate the improvement in latent variable estimation and the improvements in associated downstream tasks by simulation studies and an application to a Facebook data example.

Lastly, we extend statistical modeling from a single network to multiple networks. Entities often interact with each other through multiple types of relations, which can be represented as multilayer networks. Multilayer networks among the same set of nodes usually share common structures, while each layer can also possess its distinct node connecting behaviors. To capture such characteristics, we propose a flexible latent space model, where we embed each node with a latent vector shared among layers and a layer-specific effect for each layer, and let both elements together with a layer-specific connectivity matrix to determine edge formations. We establish theoretical properties of the maximum likelihood estimators and show that the upper bound of the common latent structure's estimation error is inversely proportional to the number of layers under mild conditions. The superior performance of the proposed model is demonstrated through simulation studies and applications to two real-world data examples.

CHAPTER I

Introduction

Network data describe the connectivity relationship among individuals and are prevalent in many scientific and engineering fields, such as social media, neuroscience, and computer science (*Newman*, 2010; *Kolaczyk and Csárdi*, 2014). Network data are composed of nodes and edges, where a node represents an individual and an edge between two nodes captures their specific type of relationship. The connections among individuals induce complex dependencies between data points, making the study of network-linked data more challenging in comparison to the classical independent data points setting. Over the decades, there has been rich amount of research on modeling and analyzing network data. The examples include but are not limited to: network modeling (see *Goldenberg et al.* (2010) for a review), community detection (*Holland et al.*, 1983; *Airoldi et al.*, 2008; *Karrer and Newman*, 2011; *Newman and Girvan*, 2004; *Newman*, 2006; *Qin and Rohe*, 2013), link predictions (*Leicht et al.*, 2006; *Liben-Nowell and Kleinberg*, 2007; *Doppa et al.*, 2009; *Kashima et al.*, 2009; *Lü and Zhou*, 2011; *Zhao et al.*, 2017; *Chen et al.*, 2019a), etc. This thesis continues studying problems of interests on network-linked data, with a special focus on the network data that are of more complex structure rather than a single network.

In the modern big data era, network-linked data are often collected with additional node variables. For example, in a social network, besides the friendship links

among people, individual characters such as age, gender, educational institution will be recorded as well (*Christakis and Fowler*, 2007; *Leskovec and Mcauley*, 2012). When the node variables are available and the focus is on the node variable side, e.g., predicting one response variable of interest, we could ask the question that how the additional network information, together with other predictors, can be incorporated into prediction models to improve the prediction performance (*Manski*, 1993; *Zhu et al.*, 2003; *Sen et al.*, 2008; *Bramoullé et al.*, 2009; *Tang et al.*, 2013; *Li et al.*, 2019). Correspondingly, if the focus is on studying network itself, we are also interested in how the additional node variables that have correlation with the network can potentially help to estimate the network structure (*Kim and Leskovec*, 2012; *Zhang et al.*, 2016; *Binkiewicz et al.*, 2017).

In some applications, the data goes beyond a single network in the sense that individuals can be interacting with each other through more than one types of relation (*Lazega et al.*, 2001; *Banerjee et al.*, 2013), therefore multiple networks among the same set of nodes, as known as multilayer networks, are collected. The tools for analyzing a single network can be utilized for studying multilayer networks, e.g., analyzing each layer separately or aggregating multiple layers into a single one. However, this may lose substantial information since multilayer networks usually share common structures among layers, meanwhile each layer would possess its own specific connecting patterns. It is still necessary to develop tailored statistical tools for multilayer network such that both the commonality and speciality can be captured (*Han et al.*, 2015; *Paul and Chen*, 2015, 2020; *De Bacco et al.*, 2017; *Levin et al.*, 2017; *Wang et al.*, 2017c; *Nielsen and Witten*, 2018; *Arroyo et al.*, 2019; *Gollini and Murphy*, 2016; *Salter-Townshend and McCormick*, 2017; *DAngelo et al.*, 2019).

This dissertation studies the above problems respectively. In chapter 2, we consider the problem of (semi-) supervised learning on network linked data, i.e., predicting one response variable of interest using both predictor and network information.

Classical prediction models usually make the assumption that data points are independent sample (*Friedman et al.*, 2001). However, when data points are connected by a network, they may interact in complex ways therefore the independence assumption might not hold. We consider incorporating network information into modeling the distribution of all variables associated with each node using matrix variate models, allowing for two-way dependence among the data points and among the variables. Further, we derive the conditional distribution of the response variable given the predictors under the specified model, and obtain a prediction model where the response of one data point depends on the covariates of all the data points in the network in a principled way. This work introduces a novel prediction framework for network-linked data, and demonstrates promising performance on numerical studies.

The main focus of the first project is on predicting the variable of interest, where the network link is treated as supplementary information to assist prediction and is viewed as fixed. In chapter 3, we shift the focus to the network itself and consider the problem of modeling network data with additional (high-dimensional) node variables. We utilize the tool of network latent space models (*Hoff et al.*, 2002; *Hoff*, 2003; *Ma and Ma*, 2017), that assume the edge formations in a network are determined by nodes individual latent positions in an unobserved space, to model the randomness in the network links. In this chapter, we propose a novel joint modeling framework, where we assume that shared latent variables determine the distribution of network and node variables simultaneously. Such modeling framework addresses the limitations in existing latent space models for handling high dimensional node variables. To fit the model, we develop an efficient projected gradient descent algorithm. We further provide theoretical guarantees on how the node variables information could be utilized to improve the estimation of latent variables associated with individual nodes. The numerical studies further support our findings.

In chapter 4, we introduce a flexible and interpretable latent space model for

modeling multilayer network data. As mentioned, multilayer networks among the same set of nodes usually share common information, while each layer can also possess its distinct node connecting patterns. Existing work (*Wang et al.*, 2017c; *Arroyo et al.*, 2019; *Valles-Catala et al.*, 2016; *DAngelo et al.*, 2019) build models for multilayer network data using network models for a single network as blocks. However, due the specific model assumption, most of the work could not accommodate enough differences among different layers. To better capture the observed characteristics on multilayer network data, we embed each node with a latent vector shared among layers and a layer-specific effect for each layer; both elements determine edge formations, together with a layer-specific connectivity matrix. We further investigate whether leveraging multilayer network information could assist the estimation of layer-shared structure, in comparison to the estimation using each network separately. We show in theory that the upper bound of the common latent structures estimation error is inversely proportional to the number of layers under mild conditions. The proposed flexible model is shown to fit the real-world data well.

Throughout this dissertation, we represent each network composed of n nodes as a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{1, \ldots, n\}$ is the node set and $\mathcal{E}$ is the edge set, with $(i, i') \in \mathcal{E}$ indicating node i and node i' are linked. The graph $\mathcal{G}$ is usually expressed as an adjacency matrix, i.e., $A \in \{0, 1\}^{n \times n}$, such that $A_{ii'} = A_{i'i} = 1$ if the pair of nodes $(i, i') \in \mathcal{E}$ and $A_{ii'} = 0$ otherwise. When multiple network exists, we represent the multilayer network as a collection of the graphs or adjacency matrices. The network A could be either viewed as fixed or random, depending on its role in the specific statistical learning and statistical modeling tasks.

CHAPTER II

Prediction for Network-linked Data using Matrix Variate Models

2.1 Introduction

Classical statistical learning problems, such as regression, classification and clustering, usually assume the training data points are independent samples. In recent years, more and more real-world applications involve individuals connected by a network. Examples include the social network, where people are connected by friendship or social interactions, or the World Wide Web, where webpages are linked by hyperlinks among them. Such connectivities make observations linked by a network dependent in complex ways. For example, one of the findings from the nationwide Framingham Heart Study [1] has shown that biological and behavioral traits of obesity are correlated across social ties (*Christakis and Fowler*, 2007), i.e., weight of a person is correlated with weight of his or her friends, siblings and spouse. Therefore, for observations connected in a network, the standard independence assumption may not hold.

Certain classical statistical frameworks have been extended to the analysis of network-linked data. When the network causes dependency among individuals, in-

[1] `https://www.framinghamheartstudy.org/`

corporating network information into modeling and algorithms is expected to help the task achieve better performance, in comparison to the methods that do not taking it into consideration. For unsupervised learning problems, network structure is often used together with covariates for clustering of data points, which is known as community detection. See *Yang et al.* (2013), *Binkiewicz et al.* (2017), *Zhang et al.* (2016) for examples of representative methods. For supervised learning, the problem of interest is predicting one response variable, using predictors as well as network information, with the expectation of improved prediction performance (*Li et al.*, 2019; *Christakis and Fowler*, 2007; *Fowler and Christakis*, 2008).

This paper focuses on the supervised learning task of predicting one variable of interest using both predictors and network structure. We consider the setting that a sample of data points are connected by a network, and each data point is associated with a vector of variables, including both predictors and a response variable. Our goal is to develop an interpretable and flexible prediction model which captures the network-induced interactions among these data points. Specifically, we propose a novel matrix variate model, which pertains meaningful interpretation of the statistical parameters, to model the distribution of the data matrix associated with network-linked observations. The relationship between the response variable and predictors can be naturally derived under the proposed framework. In addition, prediction of the variable of interest can be made under various settings, using both covariates and network link information. Our main contributions are summarized in the following subsection.

2.1.1 Main Contributions

Our first contribution is the modeling framework. Classical statistical learning methods usually model the vectors associated with individual data points as independent samples drawn from a multivariate distribution. This assumption only allows the

dependency between different variables and ignores the potential dependency among data points. In this paper, we utilize matrix variate distributions, which allow two-way dependence among data points and among variables, to model the entire data matrix. We draw a connection between the observed network and the parameters that characterize the (conditional) independence among data points, such that the network information is incorporated into the model through an interpretable way.

Next, we utilize the matrix variate distribution framework to perform prediction by deriving the conditional distribution of the response variable given all predictors. We show that the derived prediction model allows the response of a data point to depend on the covariates of all the data points in the network in a principled way, instead of incorporating network information in ad hoc ways (*Manski*, 1993; *Bramoullé et al.*, 2009). Moreover, most work on network-linked data prediction in the existing literature focus on predicting a particular type of response variable (*Neville and Jensen*, 2000; *Taskar et al.*, 2002; *Sen et al.*, 2008). The proposed framework is adaptive for both regression and classification problems, depending on the variable of interest being continuous or categorical.

Lastly, from the computational perspective, we develop two algorithms for parameter estimation: one is an iterative EM algorithm and the other is a one-step approximation algorithm. These two algorithms address the computational challenges arising in direct parameter estimation by maximizing the likelihood, in particular, obtaining the analytical solution of inverting the Kronecker sum of two matrices, which usually does not have a closed form. We further establish consistency of estimators obtained from the proposed algorithm under mild conditions.

Our proposed method differs from existing matrix variate model based work in two aspects. First, we consider a novel Kronecker sum covariance structure rather than the commonly used Kronecker product covariance, and such covariance structure enjoys benefits in terms of parameter interpretation and model flexibility. In particular, we

will show the proposed model allows the response of one data point to depend on covariates of other data points in the network, while the matrix variate distribution with the Kronecker product covariance does not have such flexibility. Secondly, the existing literature mainly focus on modeling and/or estimating general dependency among data points and do not rely on a specific observed network. As far as we know, our paper is the first that utilizes the tool of matrix variate distributions for network data analysis.

2.1.2 Related Work

Prediction for network-linked data There have not been many general statistical methods of prediction for network-linked data, though specific applications have been considered in *Wolf et al.* (2009); *Asur and Huberman* (2010); *Newman* (2014) etc. Existing predictive models usually make specific assumptions about how the response variable of one data point depends on itself and other data points in the network. For example, the social interaction model (SIM), well-studied in econometrics, and its variants (*Manski*, 1993; *Bramoullé et al.*, 2009; *Fowler and Christakis*, 2008), incorporate social effects by modeling the mean of each individual's response as a linear combination of its covariates, weighted average of its neighbors' responses and covariates. More recently, *Li et al.* (2019) proposed a prediction model with a network cohesion assumption, i.e., linked data points behave similarly. It assumes that each data point has a distinct individual effect, represented by the intercept term in the regression model, and such individual effects are smooth over linked nodes. Both methods specify explicit forms of the linear model for the response variable. Under our proposed model, the relationship between the response variable and covariates of all data points is derived based on their joint distribution, which is more general and principled.

Matrix variate distributions The matrix variate distribution has been adopted in statistical modeling for several applications, such as genetics (*Efron*, 2009; *Hornstein et al.*, 2019), spatial-temporal data (*Huizenga et al.*, 2002; *Wackernagel*, 2013), and financial trading (*Leng and Tang*, 2012). Generally speaking, instead of modeling rows in a matrix as independent samples from a multivariate distribution, matrix variate distributions model the whole matrix and are able to capture the two-way dependence among rows and among columns. Matrix variate distributions with different covariance structures have been proposed. The most commonly used one is the Kronecker product covariance (*Efron*, 2009; *Allen and Tibshirani*, 2010, 2012; *Hornstein et al.*, 2019; *Huizenga et al.*, 2002; *Wackernagel*, 2013; *Leng and Tang*, 2012). Another line of work (*Stegle et al.*, 2011; *Kalaitzis et al.*, 2013; *Rudelson and Zhou*, 2017; *Park et al.*, 2017) models the covariance or precision matrix as the sum of Kronecker products of matrices, including Kronecker sum as a special case. Most existing work on matrix variate models focus on modeling generally dependent data points and do not rely on an observed network between data points. Moreover, they mainly consider the estimation of the (conditional) independence between variables and/or between observations, but rarely consider making predictions on test data based on the inference of matrix variate models.

2.1.3 Notations and Organization

We adopt the following setup and notations throughout the paper. Consider a data matrix consisting of n observations, and each data point i is associated with a vector of variables, including both predictors and a response variable, denoted by $Z_i \in \mathbb{R}^q$, for $i = 1, \ldots, n$. The entire data matrix is denoted by $Z_{n\times q} = [Z_1, Z_2, \ldots, Z_n]^T$. Given a general matrix $M \in \mathbb{R}^{m\times n}$, denote $M_{i\cdot}$ as its ith row and $M_{\cdot j}$ as its jth column. By default, we treat all vectors as column vectors.

The rest of the paper is organized as follows. In Section 2.2, we propose to use

matrix variate distributions with the Kronecker sum covariance to model variables associated with data points connected by a network. We show how network information is incorporated into model specification and develop efficient parameter estimation algorithms. We then consider predicting a continuous variable of interest and derive the relationship between the response variable and predictors under the proposed model. Extensions to classification problems are also addressed. In Section 2.3, we establish theoretical properties of the estimators obtained from the proposed algorithm. In Section 2.4, we conduct simulation studies under multiple settings, comparing the performance of the proposed method with benchmark methods, and also apply the method to a geographical data example. We conclude the paper with discussions on potential directions of future work in Section 2.5. The technical details are relegated to the appendix.

2.2 Proposed Methods

In this section, we first introduce a novel matrix variate model for the variables associated with network-linked data points and provide its statistical interpretation. Then, two algorithms for parameter estimation are developed. Further, we show how the proposed framework is utilized for predicting a continuous response variable under different settings. The adaption for predicting a categorical response variable is also discussed.

2.2.1 Matrix Variate Model with Kronecker Sum Covariance

Motivated by the fact that the independence assumption may not hold for data points connected by a network, we consider using matrix variate distributions (MVD), which allow dependence both between rows and between columns, to model the distribution of the data matrix Z.

One of the commonly used matrix variate distributions is the matrix normal dis-

tribution. A matrix $Z_{n\times q}$ is said to follow a matrix normal distribution

$$Z_{n\times q} \sim \mathcal{MN}(0_{n\times q}, \Sigma_{q\times q} \otimes \Phi_{n\times n}) \tag{2.1}$$

if

$$vec(Z_{n\times q}) \sim \mathcal{N}(0, \Sigma_{q\times q} \otimes \Phi_{n\times n}),$$

i.e., $vec(Z_{n\times q})$ follows a multivariate normal distribution with mean 0 and covariance $\Sigma_{q\times q} \otimes \Phi_{n\times n}$. Here $vec(Z_{n\times q})$ stacks columns of $Z_{n\times q}$ into a vector in $\mathbb{R}^{nq}$, and the Kronecker product $\Sigma_{q\times q} \otimes \Phi_{n\times n}$ of two matrices is defined as:

$$\Sigma_{q\times q} \otimes \Phi_{n\times n} \coloneqq \begin{bmatrix} \Sigma_{11}\Phi & \cdots & \Sigma_{1q}\Phi \\ \vdots & \ddots & \vdots \\ \Sigma_{q1}\Phi & \cdots & \Sigma_{qq}\Phi \end{bmatrix},$$

which is of size $nq \times nq$. We require both $\Sigma_{q\times q}$ and $\Phi_{n\times n}$ be positive definite. In general, $\Sigma_{q\times q}$ is the matrix representing the relationship among q columns and $\Phi_{n\times n}$ is the matrix characterizing dependency among n rows. Under such a matrix normal distribution, the covariance between two data points $Z_{i\cdot}$ and $Z_{i'\cdot}$ is $cov(Z_{i\cdot}, Z_{i'\cdot}) = \Phi_{ii'}\Sigma$, implying that different rows of $Z_{n\times q}$ or different data points may not be independent. For two different features j and j', we have $cov(Z_{ij}, Z_{i'j'}) = \Phi_{ii'}\Sigma_{jj'}$, therefore different features of different data points could be correlated. For data point i, we have $cov(Z_{ij}, Z_{ij'}) = \Phi_{ii}\Sigma_{jj'}$, which suggests that the covariance between two features j and j' could be different for different data points.

In this paper, we consider an alternative matrix variate model rather than the matrix normal distribution. Specifically, we consider modeling the data matrix $Z_{n\times q}$ by the matrix variate distribution with a Kronekcer sum covariance:

$$Z_{n\times q} \sim \mathcal{MN}(0_{n\times q}, \Sigma_{q\times q} \oplus \Phi_{n\times n}). \tag{2.2}$$

This is equivalent to say that $vec(Z_{n\times q})$ follows a multivariate Gaussian distribution with mean 0 and a Kronecker sum covariance $\Sigma_{q\times q} \oplus \Phi_{n\times n}$, which is defined as:

$$\begin{aligned}\Sigma_{q\times q} \oplus \Phi_{n\times n} &= \Sigma_{q\times q} \otimes I_n + I_q \otimes \Phi_{n\times n} \\ &= \begin{bmatrix} \Sigma_{11}I_n + \Phi & \Sigma_{2q}I_n & \cdots & \Sigma_{1q}I_n \\ \Sigma_{21}I_n & \Sigma_{22}I_n + \Phi & \cdots & \Sigma_{2q}I_n \\ \vdots & \vdots & \ddots & \vdots \\ \Sigma_{q1}I_n & \Sigma_{q2}I_n & \cdots & \Sigma_{qq}I_n + \Phi \end{bmatrix}.\end{aligned}$$

Under such a model, the covariance between two data points $Z_{i\cdot}, Z_{i'\cdot}$ is

$$cov(Z_{i\cdot}, Z_{i'\cdot}) = \begin{cases} \Phi_{ii'}I_q & i \neq i', \\ \Phi_{ii}I_q + \Sigma & i = i'. \end{cases}$$

Therefore, for two distinct data points, their different features would be independent. For two variables j and j', since $cov(Z_{ij}, Z_{ij'}) = \Sigma_{jj'}$ for all i, the off-diagonal elements of Σ represent the covariance between different variables. Note when $\Phi \propto I_n$, the proposed model is equivalent to the case where the rows of Z are i.i.d samples drawn from a multivariate normal distribution. In comparison to the matrix normal distribution, the Kronecker sum covariance model assumes sparsity on the correlation between different features of different data points; further, it makes the interpretation of off-diagonal elements of $\Sigma_{q\times q}$ more clear.

To incorporate the network information, we make a connection between $\Phi_{n\times n}$, the matrix describing dependency between observations, and the adjacency matrix of the network $A_{n\times n}$. Specifically, we set $\Phi^{-1}_{ii'} = 0$ whenever $A_{ii'} = 0$, for $i \neq i'$. In other words, we assume the conditional independence among data points that are not linked in the network.

Remark II.1. The proposed model has a clear interpretation as a hierarchical model.

Specifically, we could view the data matrix $Z_{n\times q}$ as the sum of two independent components: $Z_{n\times q} = W_{n\times q} + \epsilon_{n\times q}$, where $W_{n\times q}$ could be considered as the 'signal' part of data points while $\epsilon_{n\times q}$ could be seen as the 'noise' part. Given $W_{n\times q}$, we specify $Z_{n\times q}|W_{n\times q} \sim \mathcal{MN}(W_{n\times q}, \Sigma_{q\times q}\otimes I_n)$. This is equivalent to specifying the distribution of $\epsilon_{n\times q}$ as $\epsilon_{n\times q} \sim \mathcal{MN}(0, \Sigma_{q\times q}\otimes I_n)$, or the rows of $\epsilon_{n\times q}$ are i.i.d with $\epsilon_{i\cdot} \sim \mathcal{N}(0, \Sigma_{q\times q})$. In other words, given $W_{n\times q}$, the rows of $Z_{n\times q}$ are independently distributed, with each row $Z_{i\cdot}$ following the distribution $Z_{i\cdot}|W_{n\times q} \sim \mathcal{N}(W_{i\cdot}, \Sigma_{q\times q})$. Moreover, we assume $W_{n\times q} \sim \mathcal{MN}(0, I_q\otimes\Phi_{n\times n})$. This is equivalent to saying that the columns of $W_{n\times q}$ are independent but for each column $W_{\cdot j}$, different data points are correlated with $W_{\cdot j} \sim \mathcal{N}(0, \Phi_{n\times n})$. The specifications on distributions of $W_{n\times q}$ and $Z_{n\times q}|W_{n\times q}$ lead to the marginal distribution of $Z_{n\times q}$ as $Z_{n\times q} \sim \mathcal{MN}(0_{n\times q}, \Sigma_{q\times q}\otimes I_n + I_q\otimes\Phi_{n\times n})$. In addition, since $\Phi_{n\times n}$ is the covariance matrix of different rows of $W_{n\times q}$, the requirement we put on the zero elements in Φ^{-1} is essentially specifying the conditional independence between signal parts of the data points.

Remark II.2. Work independent of this paper (*Rudelson and Zhou*, 2017; *Park et al.*, 2017) have also considered modeling the covariance of matrix variate data as a Kronecker sum of two matrices. *Rudelson and Zhou* (2017) considered the problem of regression when errors in variables exist, where they specified the observed covariates follow a Kronecker sum covariance matrix variate distribution. The main focus was to recover a sparse regression coefficient vector in the regression model, but they did not address the problem of estimating Σ and Φ, as we will address in the following subsection. *Park et al.* (2017) utilized the same distribution to model spatial-temporal data and proposed a parameter estimation approach based on nodewise regression. Our paper studies a completely different problem, as the main focus is to predict a response variable based on an observed network, and we develop new parameter estimation algorithms that are different from those in the existing work.

2.2.2 Parameter Estimation

Estimating Σ and Φ directly by maximizing the marginal likelihood of $Z_{n\times q}$ is difficult as it involves the analytical solution of the inverse of Kronecker sum of two matrices, which usually does not have a closed form. In this subsection we develop two parameter estimation methods, one being an iterative EM algorithm and the other a one-step approximation algorithm. Both approaches are based on the hierarchical interpretation of the model as discussed in Remark II.1.

2.2.2.1 EM Algorithm

The first approach is to consider either W or ϵ as latent variables and estimate Σ and Φ by the standard EM algorithm. Here we view W as latent variables and derive the E-step and M-step as follows.

E-step Within each iteration of the E-step, we first calculate the log-likelihood of the complete data, $(Z_{n\times q}, W_{n\times q})$, denoted by $l_c(\Sigma, \Phi|Z, W)$. Specifically, we have

$$\begin{aligned} l_c(\Sigma, \Phi|Z, W) &= \log P(W|\Phi) + \log P(Z|W, \Sigma) \\ &= -q\log|\Phi| - tr(\Phi^{-1}WW^T) - n\log|\Sigma| - tr\big(\Sigma^{-1}(Z-W)^T(Z-W)\big). \end{aligned}$$

The distribution of latent variables conditional on observed data, $W|Z$, is given by

$$W|Z \sim \mathcal{MN}(\mu_W, \Sigma_W),$$

where

$$\Sigma_W = \Omega_W^{-1} = \big(I_q \otimes \Phi^{-1} + \Sigma^{-1} \otimes I_n\big)^{-1}, \tag{2.3}$$

and

$$vec(\mu_W) = \Sigma_W\Big((\Sigma^{-1} \otimes I_n)vec(Z)\Big). \tag{2.4}$$

Then we take expectation of $l_c(\Sigma, \Phi|Z, W)$ with respect to $W|Z$, obtaining the quantity $E_{W|Z} l_c(\Sigma, \Phi|Z, W)$ and getting ready for the M-step.

M-step In the M-step, we solve for $\widehat{\Sigma}$ and $\widehat{\Phi}$ by maximizing $\mathbb{E}_{W|Z}\left(l_c(\Sigma, \Phi|Z, W)\right)$. There are several issues that need attention in the M-step. First, in the Kronecker sum $\Sigma_{q\times q} \oplus \Phi_{n\times n}$, $\Sigma_{q\times q}$ and $\Phi_{n\times n}$ are not identifiable since we can add a constant to all diagonal elements of one matrix and subtract the same constant from the diagonal of the other (while keeping the positive definiteness of both) and obtain the same Kronecker sum. Therefore we need to add further constraints on diagonal elements of Σ and Φ, for example, requiring trace of Φ be a known constant.

Secondly, since we are essentially using q columns of Z to estimate an $n \times n$ matrix Φ (and similarly the other way around for Σ), so when $n \gg q$, we could not afford too many parameters in $\Phi_{n\times n}$. In this situation, we let Φ take a specific form that $\Phi^{-1} \propto (L + \gamma I_n)$ for some constant $\gamma > 0$. Here L is the Laplacian of the adjacency matrix A, defined as $L = \text{diag}(d_1, \ldots, d_n) - A$, where $d_i = \sum_{i'=1}^{n} A_{ii'}$ is the degree of node i. This specification satisfies the requirement that Φ^{-1} and A share the same locations of zero elements as we have proposed in Section 2.2.1. Another situation is the high-dimensional setting where $n \gg q$. If this is the case, we assume sparsity for Σ^{-1}, and we maximize the objective quantity $\mathbb{E}_{W|Z}\left(l_c(\Sigma, \Phi|Z, W)\right)$ under such constraints. This maximization could be done by existing tools, such as Graphical Lasso (*Friedman et al.*, 2008).

Note that (2.3) and (2.4) in E-step involve numerically inverting Kronecker sum of two matrices. Given two matrices $\Sigma_{q\times q}$ and $\Phi_{n\times n}$, their Kronecker sum is of size $nq \times nq$ and directly computing the inverse of it can be computationally expensive. We come up with an efficient computational solution. Let $\Sigma = U_1 \Lambda_1 U_1^T$ and $\Phi = U_2 \Lambda_2 U_2^T$

be their eigen-decompositions respectively, then

$$\Sigma \otimes I_n + I_q \otimes \Phi = (U_1 \Lambda_1^{1/2} \otimes U_2)(I_q \otimes I_n + \Lambda_1^{-1} \otimes \Lambda_2)(\Lambda_1^{1/2} U_1^T \otimes U_2^T).$$

It is straightforward to take the inverse of the middle term since it is diagonal. The inverse of the left and right parts are also not difficult to obtain under the property of Kronecker product operation. The main computational cost is now due to eigen-decomposition of matrices, and therefore the computational complexity is reduced from $O(n^3 q^3)$ to $O(n^3 + q^3)$. For very large n or q, we could also consider partial eigen-decompositions of a matrix to accelerate the computation.

We summarize main steps of the EM algorithm in Algorithm 1. Detailed derivations of the EM algorithm, including efficient computational strategies, are postponed to the Appendix.

Algorithm 1 EM algorithm for estimating MVD with Kronecker sum covariance

1: Input: data matrix $Z_{n\times q}$, network adjacency matrix $A_{n\times n}$, hyperparameters for graphical lasso and identifiability condition.
2: **Initialization**
3: Initialize Σ by solving graphical lasso, with $Z^T Z/n$ as input sample covariance matrix.
4: Initialize Φ by solving graphical lasso under appropriate constraints, with ZZ^T/q as input sample covariance matrix.
5: **repeat**
6: **E-step**: Calculate $E_{W|Z} l_c(\Sigma, \Phi | Z, W)$.
7: **M-step**: Update Σ, Φ by maximizing $E_{W|Z} l_c(\Sigma, \Phi | Z, W)$ under appropriate constraints; project Σ, Φ such that they satisfy the identifiability condition.
8: **until** Convergence of log-likelihood of $P(Z|A)$
9: Output: $\widehat{\Sigma}, \widehat{\Phi}$

Remark II.3. The above algorithm describes the case when only one data matrix Z is available. In some cases, we may have multiple data matrices (e.g. the data example in Section 2.4.3). The proposed EM algorithm can be naturally extended to the case with multiple data matrices, under the assumption that all data matrices are i.i.d from the same matrix variate distribution. The details are omitted.

2.2.2.2 Approximation Algorithm

Recall we can view Z as $Z = W + \epsilon$, where $W \sim \mathcal{MN}(0, I_q \otimes \Phi)$ and $\epsilon \sim \mathcal{MN}(0, \Sigma \otimes I_n)$ are independent. Assuming the counterfactual case that latent variables W and ϵ are observed, then we could estimate $\Omega_0 = \Sigma^{-1}$ and $\Theta_0 = \Phi^{-1}$ using graphical lasso by solving the following two problems:

$$\widehat{\Omega}_\lambda = \arg\min_{\Omega \succeq 0} \left\{ \mathrm{tr}\left(\Omega \widetilde{\Sigma}\right) - \log|\Omega| + \lambda \|\Omega\|_{1,\mathrm{off}} \right\}, \tag{2.5}$$

$$\widehat{\Theta} = \arg \min_{\substack{\Theta \succeq 0 \\ \Theta_{ij}=0, (i.j) \notin \mathcal{E}}} \left\{ \mathrm{tr}\left(\Theta \widetilde{\Phi}\right) - \log|\Theta| \right\}, \tag{2.6}$$

where $\widetilde{\Sigma} = \epsilon^T \epsilon / n$ and $\widetilde{\Phi} = WW^T / q$ are sample covariances, and $\|\Omega\|_{1,\mathrm{off}}$ refers to sum of absolute value of all off-diagonal elements of Ω. However, since W and ϵ are not directly observable, we consider first approximating $\widetilde{\Sigma} = \epsilon^T \epsilon / n$ and $\widetilde{\Phi} = WW^T / q$ using observed Z and then plugging them into (2.5) and (2.6) to obtain the estimators. To achieve this, note that

$$\mathbb{E}\left(\frac{Z^T Z}{n}\right) = \mathbb{E}\left(\frac{W^T W}{n}\right) + \mathbb{E}\left(\frac{\epsilon^T \epsilon}{n}\right) = \frac{\mathrm{tr}(\Phi)}{n} I_q + \mathbb{E}\left(\frac{\epsilon^T \epsilon}{n}\right) = \frac{\mathrm{tr}(\Phi)}{n} I_q + \Sigma.$$

Therefore, we could use $\widehat{\Sigma} = Z^T Z / n - \widehat{\mathrm{tr}}(\Phi) I_q / n$ as an approximation to $\widetilde{\Sigma} = \epsilon^T \epsilon / n$, where $\widehat{\mathrm{tr}}(\Phi)$ is an estimate of $\mathrm{tr}(\Phi)$. It is possible that $\widehat{\Sigma}$ obtained in this way is not semi-positive definite (SPD). If so, we project it to a $q \times q$ SPD matrix. Similarly we can use $\widehat{\Phi} = ZZ^T / q - \widehat{\mathrm{tr}}(\Sigma) I_n / q$ as an approximation to $\widetilde{\Phi}$, where $\widehat{\mathrm{tr}}(\Sigma)$ is an estimate of $\mathrm{tr}(\Sigma)$. Moreover, due to the identifiability issue of the diagonal elements of Σ and Φ, we assume one of the two traces is known. For example, when assuming $tr(\Phi)$ is known, we have $\widehat{\mathrm{tr}}(\Sigma) = \left(\|Z\|_F^2 - q\mathrm{tr}(\Phi)\right)_+ / n$. Then we consider estimating $\Omega_0 = \Sigma^{-1}$ and $\Theta_0 = \Phi^{-1}$ by solving:

$$\widehat{\Omega}_\lambda = \arg\min_{\Omega \succeq 0} \left\{ \mathrm{tr}\left(\Omega \widehat{\Sigma}\right) - \log|\Omega| + \lambda \|\Omega\|_{1,\mathrm{off}} \right\}, \tag{2.7}$$

$$\widehat{\Theta} = \arg \min_{\substack{\Theta \succeq 0 \\ \Theta_{ij}=0, (i.j) \notin \mathcal{E}}} \left\{ \mathrm{tr}\left(\Theta \widehat{\Phi}\right) - \log |\Theta| \right\}. \tag{2.8}$$

We call this algorithm an approximation algorithm since we use approximated sample covariances of W and ϵ. The details are summarized in Algorithm 2. In comparison to the iterative EM algorithm, the approximation algorithm could be done in one step without iterations, therefore it is computationally more efficient. We establish theoretical properties for the estimators from the approximation algorithm in Section 2.3 and compare numerical performances of both algorithms in Section 2.4.

Algorithm 2 Approximation algorithm for estimating MVD with Kronecker sum covariance

1: Input: data matrix $Z_{n\times q}$, network adjacency matrix $A_{n\times n}$, hyperparameters for graphical lasso and identifiability condition
2: Calculate $\widehat{\Sigma} = Z^T Z/n - tr(\Phi) I_q/n$; project $\widehat{\Sigma}$ to a SPD matrix.
3: Calculate $\widehat{\Phi} = ZZ^T/q - \widehat{\mathrm{tr}}(\Sigma) I_n/q$, where $\widehat{\mathrm{tr}}(\Sigma) = \left(\|Z\|_F^2 - q\mathrm{tr}(\Phi)\right)_+ /n$. Project $\widehat{\Phi}$ to a SPD matrix and rescale $\widehat{\Phi}$ so that its trace equals to the pre-specified value.
4: Obtain $\widehat{\Omega}_\lambda$ by (2.7)
5: Obtain $\widehat{\Theta}$ by (2.8)
6: Output: $\widehat{\Omega}_\lambda$ and $\widehat{\Theta}$

2.2.3 Regression on Network-linked Data

The previous section introduces a general matrix variate distribution framework to model the variables associated with each node in a network, which allows dependence among data points. In this subsection, we show how to perform regression on network-linked data using the proposed framework.

To adapt the proposed model to the regression setting, we distinguish the variables associated with nodes by the response variable and predictors. Specifically, we assume each node i is associated with $q = (p+1)$ variables: $(X_i, Y_i) \in \mathbb{R}^{p+1}$, where $X_i \in \mathbb{R}^p$ is the vector of covariates and $Y_i \in \mathbb{R}$ is the response variable. We let $X_{n\times p} = [X_1, X_2, \ldots, X_n]^T$ be the covariates matrix and $Y = (Y_1, Y_2, \ldots Y_n) \in \mathbb{R}^n$ be the response vector. The entire data matrix is given by $Z_{n\times(p+1)} = (X_{n\times p}, Y_{n\times 1})$.

Based on the model assumption in Section 2.2.1, we have

$$Z_{n\times(p+1)} \sim \mathcal{MN}(0, \Sigma_{(p+1)\times(p+1)} \oplus \Phi_{n\times n}).$$

Denote $\Sigma = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix}$, where $\Sigma_{xx} \in \mathbb{R}^{p\times p}$, $\Sigma_{yx}^T = \Sigma_{xy} \in \mathbb{R}^{p\times 1}$ and $\Sigma_{yy} \in \mathbb{R}$. Then by standard multivariate normal theory, we could derive the distribution of $Y_{n\times 1}$, conditional on all predictors $X_{n\times p}$ as:

$$\begin{aligned} Y|vec(X) &\sim \mathcal{N}(\Sigma_{yx} \otimes I_n \left(\Sigma_{xx} \otimes I_n + I_p \otimes \Phi\right)^{-1} vec(X), \\ (\Sigma_{yy} I_n + \Phi) &- (\Sigma_{yx} \otimes I_n)(\Sigma_{xx} \otimes I_n + I_p \otimes \Phi)^{-1}(\Sigma_{xy} \otimes I_n)). \end{aligned} \tag{2.9}$$

To understand the flexibility of the relationship between the response variable and predictors under the proposed matrix variate model, we first go back to the case when the observations $\{(X_i, Y_i)\}$ are i.i.d. samples from $\mathcal{N}(0, \Sigma_{(p+1)\times(p+1)})$. In this case, the conditional distribution of Y_i, given all the other variables, is:

$$Y_i|X \sim \mathcal{N}(\Sigma_{yx}\Sigma_{xx}^{-1}X_i, \Sigma_{yy} - \Sigma_{yx}\Sigma_{xx}^{-1}\Sigma_{xy}).$$

This is equivalent to a multiple linear regression model, where $Y_i = \beta^T X_i + \epsilon_i$ for $i = 1, 2, \ldots, n$, with $\beta^T = \Sigma_{yx}\Sigma_{xx}^{-1}$ and ϵ_i's being i.i.d normal random variables with mean 0 and variance $\Sigma_{yy} - \Sigma_{yx}\Sigma_{xx}^{-1}\Sigma_{xy}$. Note due to the independence assumption, the response of data point i only depends on its own covariates X_i.

As a comparison, based on (2.9), we have

$$\mathbb{E}(Y|vec(X)) = \Sigma_{yx} \otimes I_n \left(\Sigma_{xx} \otimes I_n + I_p \otimes \Phi\right)^{-1} vec(X).$$

Therefore, $\mathbb{E}(Y_i|X)$ involves not only X_i but also other $X_{i'}$'s for $i' \neq i$, due to the complex structure of the matrix $\Sigma_{yx} \otimes I_n \left(\Sigma_{xx} \otimes I_n + I_p \otimes \Phi\right)^{-1}$ multiplied in front

of $vec(X)$. In other words, to predict the response variable of node i, we are using information not limited to covariates of the ith node. Such relationship between Y_i and X is more flexible than a linear model of the form $Y_i = \alpha_i + \beta^T X_i + \epsilon_i$ as in *Li et al.* (2019). Further, the proposed way of incorporating covariates of other data points is statistically principled, instead of ad hoc feature engineering methods such as aggregating neighbors' covariates (*Bramoullé et al.*, 2009).

This flexible conditional distribution in (2.9) brings out another motivation of choosing the Kronecker sum covariance over the matrix normal distribution with Kronecker product covariance. Consider the situation where the data matrix $Z_{n\times(p+1)}$ is modeled by a matrix normal distribution

$$Z_{n\times(p+1)} = (X_{n\times p}, Y_{n\times 1}) \sim \mathcal{MN}(0, \Sigma_{(p+1)\times(p+1)} \otimes \Phi_{n\times n}).$$

Then it is not difficult to derive that

$$Y|vec(X) \sim \mathcal{N}(\Sigma_{yx}\Sigma_{xx}^{-1} \otimes I_n vec(X), \left(\Sigma_{yy} - \Sigma_{yx}\Sigma_{xx}^{-1}\Sigma_{xy}\right) \otimes \Phi).$$

Note the conditional mean part takes the same form as in the i.i.d case. For some non-iterative estimation method for the matrix normal distribution such as *Zhou* (2014), the estimation of Σ depends on Φ only up to a multiplicative scale, which does not play a role in the conditional mean part, as it would be cancelled by the coefficient vector $\Sigma_{yx}\Sigma_{xx}^{-1}$. Therefore, regardless of the assumption on the dependence among data points, we would obtain the same conditional expectation of $Y|X$ as in the i.i.d case. It is thus less flexible than the Kronecker sum covariance model in incorporating information in the network when making predictions, even though the conditional covariance of $Y|vec(X)$ allows dependence among data points.

Based on the derived relationship between the response variable and predictors, we now consider making predictions under two scenarios.

In-sample prediction We first consider the case that predictions are made on the same samples. For example, for an online payment platform, given the transaction network and expenses of the current month, we are interested in predicting the balance of each user in the next month. For in-sample prediction, we assume the training data $Z_{n\times(p+1)} = (X_{n\times p}, Y_{n\times 1})$ and the network A are fully observed. The new testing data $Z^*_{n\times(p+1)} = (X^*_{n\times p}, Y^*_{n\times 1})$ are generated by the same matrix variate distribution, on the same set of nodes, but only $X^*_{n\times p}$ is observed and we need to predict $Y^*_{n\times 1}$. Under this setting, we first estimate Φ and Σ using the training data via algorithms described in Section 2.2.2. Then we predict Y^* by its conditional expectation given X^*, i.e., $\widehat{\Sigma}_{yx} \otimes I_n \left(\widehat{\Sigma}_{xx} \otimes I_n + I_p \otimes \widehat{\Phi}\right)^{-1} vec(X^*)$ with plug-in estimators.

Semi-supervised learning Another common situation is semi-supervised learning. Examples of semi-supervised learning prediction are common in online social networks, where for example, the friendship network is observed but some node features of interest may be partially observed. Those missing values could be predicted by other fully-observed covariates and the network. In this setting, we assume the design matrix $X_{n\times p}$ and the network A are fully observed, however the response variable Y is only partially observed. We denote $Y_{n\times 1} = (Y_L^T, Y_U^T)^T$, where Y_L are observed and Y_U are unobserved. Since Φ, the matrix that describes covariance among data points are shared across different variables, we could first estimate $\Phi_{n\times n}$ by fully observed columns, i.e., $X_{n\times p}$. Similarly we could estimate $\Sigma_{(p+1)\times(p+1)}$ by fully observed rows $(X_L, Y_L) \in \mathbb{R}^{|L|\times(p+1)}$, where X_L correspond to rows of X that Y is observed. Finally we predict unobserved Y, Y_U by

$$\left[\widehat{\Sigma}_{yx} \otimes I_n \left(\widehat{\Sigma}_{xx} \otimes I_n + I_p \otimes \widehat{\Phi}\right)^{-1} vec(X)\right]_U.$$

2.2.4 Classification Setting

In this section we extend this prediction framework to the case when the variable of interest is categorical. Assume for each node i, in addition to the vector of covariates $X_i \in \mathbb{R}^p$, another variable $Y_i \in \{1, 2, \ldots, K\}$ that represents the class label is also available. Here K is the number of classes. Unlike in the regression problem, where the joint distribution of predictors X and response variable Y is modeled by a matrix variate distribution, for classification problem we consider using the class labels Y to help specify the mean structure of the distribution of X. Specifically, we assume given Y, the distribution of $X|Y$ follows a matrix variate distribution with a non-zero mean and a Kronecker sum covariance:

$$X_{n\times p}|Y_{n\times 1} \sim \mathcal{MN}(M_{n\times p}, \Sigma_{p\times p} \oplus \Phi_{n\times n}), \tag{2.10}$$

where the mean $M_{n\times p} = C\mu$, with C being an $n \times K$ matrix and $C_{ik} = \mathbf{1}(Y_i = k)$, and $\mu_{K\times p} = [\mu_1^T, \mu_2^T, \ldots, \mu_K^T]^T$, with $\mu_k \in \mathbb{R}^p$ being a row vector representing the mean parameter for class k. For the distribution of $Y_1, Y_2, \ldots, Y_n$, we consider the simple case where Y_i's are i.i.d multinomial with parameter $\pi = (\pi_1, \ldots, \pi_K)$ and $\sum_{k=1}^K \pi_k = 1$.

The parameters to be estimated are π, μ, Σ, and Φ. We could estimate π by its MLE: $\hat{\pi}_k = \sum_{i=1}^n \mathbf{1}(Y_i = k)/n$. The estimation of Φ and Σ could be achieved by the EM algorithm, but it needs modifications since the mean of $X|Y$ is in general non-zero and updates of Φ and Σ would depend on the value of M (or μ). For the estimation of μ, note if we take derivative of the log-likelihood of X given Y based on (2.10) with respect to $vec(\mu)$ and set it to zero, we have:

$$\begin{aligned} vec(\mu) =& \Big((I_p \otimes C)^T (I_p \otimes \Phi + \Sigma \otimes I_n)^{-1} (I_p \otimes C)\Big)^{-1} \\ & (I_p \otimes C)^T (I_p \otimes \Phi + \Sigma \otimes I_n)^{-1} vec(X). \end{aligned} \tag{2.11}$$

We consider two procedures for parameter estimation. The first is an iterative approach, updating μ by equation (2.11) after each EM iteration of updating Σ and Φ. The second is a two-step approach. We first center the design matrix X within each class. Then we could estimate Φ and Σ as in the case when mean is 0, and lastly update μ by equation (2.11) after obtaining $\widehat{\Phi}$ and $\widehat{\Sigma}$.

Since the above specifications require the class labels Y be fully observed, we focus on the in-sample prediction for the classification problem.

In-sample prediction Assume design matrix $X_{n\times p}$, class labels $Y_{n\times 1}$ and network $A_{n\times n}$ are observed. Assume new test data $(X^*_{n\times p},\, Y^*_{n\times 1})$ are generated by the same generative model on the same set of nodes, and $Y^*_{n\times 1}$ are unobserved. To predict Y^*, we consider the criterion:

$$\widehat{Y}^* = \arg \max_{Y^* \in \{1,2,\ldots,K\}^n} P(Y^*|X^*).$$

However, obtaining $\widehat{Y}^*$ directly from this criterion is intractable since it requires enumeration of all possible K^n assignments of labels. Therefore, we estimate $\widehat{Y}^*$ by approximate inference algorithms, e.g., variational methods (*Jordan et al.*, 1999). Due to the space constraint, we leave the derivation of the variational inference and more details about parameter estimation for classification to the Appendix.

2.3 Theoretical Properties

In this section, we show the solutions of the approximation algorithm in (2.7) and (2.8) can consistently estimate $\Omega_0 = \Sigma^{-1}$ and $\Theta_0 = \Phi^{-1}$ respectively under mild conditions.

For any matrix $M \in \mathbb{R}^{p\times p}$, we denote $\varphi_{\max}(M)$ and $\varphi_{\min}(M)$ as its largest and smallest eigenvalues respectively. Let $\|M\|_F = \sqrt{\sum_{j,k} M_{jk}^2}$ be its Frobenious norm,

and $\|M\|_2 = \varphi_{\max}(M)$ be its operator norm. Define stable rank of a matrix as $\|M\|_F^2/\|M\|_2^2$. We make the following assumptions about Σ and Φ:

Assumption II.4. *There exist* $\underline{k}_1$ *and* $\overline{k}_1$ *such that* $0 < \underline{k}_1 \leq \psi_{\min}(\Sigma) \leq \psi_{\max}(\Sigma) \leq \overline{k}_1 < \infty$.

Assumption II.5. $tr(\Sigma) = O(q)$, *i.e.,* $\exists 0 < \underline{c}_1 < \overline{c}_1 \leq \infty$, *such that* $\underline{c}_1 q < tr(\Sigma) < \overline{c}_1 q$.

Assumption II.6 (Rank condition for Φ)**.** $\|\Phi\|_F^2/\|\Phi\|_2^2 \geq \log q$.

Assumption II.4 from *Rothman et al.* (2008) guarantees that Σ^{-1} exists. Since Σ is of size $q \times q$, Assumption II.5 suggests that the diagonal elements of Σ, or the variance among variables, are of a constant order. The requirement in Assumption II.6 on $\|\Phi\|_F^2/\|\Phi\|_2^2$ suggests that the eigenvalues of Φ should not vanish too fast. Under Assumptions II.4-II.6, we can obtain an error bound for estimators obtained from (2.7) as stated in the following theorem.

Theorem II.7. *Let* $\widehat{\Omega}_\lambda$ *be the solution of* (2.7)*. Under Assumptions II.4-II.6, if* $\lambda = O(\log q/n)$*, we have*

$$\|\widehat{\Omega}_\lambda - \Omega_0\|_F = O_P\left(\sqrt{\frac{(q+s)\log q}{n}}\right),$$

where the number of non-zero entries in Ω_0 *is bounded by* s*.*

The result shows that $\widehat{\Omega}_\lambda$ consistently estimates Ω_0 in the Frobenious norm. The convergence rate is the same as that obtained by *Rothman et al.* (2008), when data points i.i.d drawn from multivariate Gaussian $\mathcal{N}(0, \Sigma)$ are observed. The result demonstrates that even the i.i.d $\mathcal{N}(0, \Sigma)$ error terms ϵ are not directly observed, we could use the observed data matrix Z to approximate its sample covariance $\epsilon^T\epsilon/n$ and achieve the same error bound for estimating Σ^{-1}. On the technical side, we use

the results in *Rudelson and Zhou* (2017) to control the maximum value of $|\widehat{\Sigma} - \Sigma|$ and the framework in *Rothman et al.* (2008) to establish the consistency of Ω_λ. Details are relegated to the Appendix.

Similarly, we need the following conditions to establish consistency for estimating Φ.

Assumption II.8. *There exist* $\underline{k}_2$ *and* $\leq \overline{k}_2$ *such that* $0 < \underline{k}_2 \leq \psi_{\min}(\Phi) \leq \psi_{\max}(\Phi) \leq \overline{k}_2 < \infty$.

Assumption II.9. $\Phi_{ii} = c_\Phi$ *for* $i = 1, 2, \ldots, n$, *where* c_Φ *is a known constant.*

Assumption II.10 (Rank condition for Σ)**.** $\|\Sigma\|_F^2 / \|\Sigma\|_2^2 \geq \log n$.

Note that we require the trace of Φ be known mainly for identifiability reason. Assumption II.9 is a special case of Φ with known trace. Based on Assumptions II.8-II.10, we have the following result about the estimation of Θ_0.

Theorem II.11. *Let* $\widehat{\Theta}$ *be the solution in* (2.8). *Under Assumptions II.8-II.10, we have*

$$\|\widehat{\Theta} - \Theta_0\|_F = O_P\left(\sqrt{\frac{2E \log n}{p}}\right),$$

where $E = |\mathcal{E}|$ *is the number of edges in the observed network* $\mathcal{G}$.

Note that in the result of Theorem II.7, the term $\sqrt{q \log q / n}$ comes from estimating the diagonal elements of Σ. However by Assumption II.9 where we assume all diagonal elements of Φ are a known constant, we could get rid of the term $\sqrt{n \log n / q}$ and obtain the rate in Theorem II.11.

Proofs of Theorems II.7 and II.11 are provided in Appendix.

2.4 Numerical Studies

In this section, we apply the methods in Section 2.2.2 to synthetic data generated from the proposed matrix variate model and a real geographic network data example.

We compare the performance of matrix variate distribution (MVD) based methods with benchmark methods.

2.4.1 Simulation Studies: Regression

Parameter settings We consider regression for both low dimensional ($n = 200$, $p = 2$) and high dimensional ($n = 50$, $p = 19$) settings. When $n = 200$, $p = 2$, we first generate a network by Erdős-rényi random graph $\mathcal{G}(n, 0.05)$. Then we set $\Phi = \tau^2(L + \gamma I_n)^{-1}$, where L is the graph Laplacian. We set $\Sigma_{(p+1)\times(p+1)}$ as

$$\Sigma = \begin{bmatrix} 1 & 0 & 0.5 \\ 0 & 1 & 0.5 \\ 0.5 & 0.5 & 1 \end{bmatrix}.$$

When $n = 50$, $p = 19$, we generate the network by the stochastic block model (SBM) with connection probability matrix

$$\begin{bmatrix} 0.15 & 0.01 \\ 0.01 & 0.15 \end{bmatrix},$$

where the nodes are randomly divided into two communities with equal probability. We first get Φ_0 by $\Phi_0 = (L + 0.1I)^{-1}$, then rescale Φ_0 to be a correlation matrix and followed by multiplying a constant c_Φ, where c_Φ represents the scale of Φ. We consider two settings of Σ:

AR(1): Set $\Sigma_{ij} = \rho^{|i-j|}$ where $\rho = 0.9$. Under this setting, Σ^{-1} is trigonal and sparse.

Independent predictors: We let the upper left $p \times p$ block of Σ be an identity matrix I_p, and set entries of the last column and the last row of Σ as $1/\sqrt{p}$. We add a small constant to the diagonal of Σ to guarantee its positive definiteness. Under this setting, different covariates are independent and each covariate is only correlated

with Y. The precision matrix Σ^{-1} is not sparse in this case.

Generate data matrix Note that the covariance $\Sigma \oplus \Phi$ could be written as

$$\Sigma_{(p+1)\times(p+1)} \otimes I_n + I_{p+1} \otimes \Phi_{n\times n} = (U_1\Lambda_1^{1/2} \otimes U_2)(I_p \otimes I_n + \Lambda_1^{-1} \otimes \Lambda_2)(\Lambda_1^{1/2}U_1^T \otimes U_2^T),$$

where $\Sigma = U_1\Lambda_1U_1^T$ and $\Phi = U_2\Lambda_2U_2^T$ are eigen-decompositions. To generate the data matrix $Z_{n\times(p+1)} = (X_{n\times p}, Y_{n\times 1})$, we first generate a $n \times (p+1)$ matrix Z_0 with each entry being i.i.d $\mathcal{N}(0,1)$, i.e., $vec(Z_0) \sim \mathcal{N}(0, I_{p+1} \otimes I_n)$. Then the data matrix $Z = (U_1\Lambda_1^{1/2} \otimes U_2)\sqrt{I_p \otimes I_n + \Lambda_1^{-1} \otimes \Lambda_2}Z_0$ follows the matrix variate distribution with mean 0 and covariance $\Sigma \oplus \Phi$. For each generated data matrix, we center it within each column to make the mean of each variable be 0.

When $n \gg p$, we compare the result of the EM algorithm with that of the ordinary least squares (OLS). When $n \not\gg p$, we compare the results of the iterative EM algorithm, the one-step approximation algorithm and the prediction procedure when samples are assumed to be i.i.d. As mentioned in Section 2.2.3, given a data matrix $Z_{n\times(p+1)}$, if the rows are i.i.d sampled from $\mathcal{N}(0, \Sigma_{(p+1)\times(p+1)})$, then for each i, we have

$$\mathbb{E}(Y_i|X_i) = \Sigma_{yx}\Sigma_{xx}^{-1}X_i. \tag{2.12}$$

Therefore, as a benchmark method, we first estimate Σ by graphical lasso, then we obtain predicted response by equation (2.12) with plug-in estimators.

Results We evaluate the performance of different methods by the mean square error (MSE), estimated by $\|\widehat{Y} - \mathbb{E}(Y|X)\|^2/n$. The performance is evaluated on test datasets that is generated by the same distribution of the training data. The results are replicated over 30 times.

Figure 2.1 shows how the performance of the proposed method on in-sample prediction in the low-dimensional setting is influenced by parameters τ^2 and γ in

$\Phi = \tau^2(L + \gamma I)^{-1}$. In the left panel with varying τ^2 and fixed $\gamma = 1$, when τ^2 is small, the effect of Φ is almost negligible, and the covariance is dominated by the term $\Sigma \otimes I_n$, therefore the rows of the data matrix can be viewed as almost n i.i.d samples. This causes the performance of the proposed EM algorithm and that of OLS similar. When τ^2 becomes larger and the dependence between data points generated by the matrix variate model gets stronger, the proposed method outperforms the OLS more significantly. In the right panel, we fix $\tau^2 = 5$ and vary the value of γ. Note that when $\Phi \propto I_n$, our model would reduce to the i.i.d case. Thus, when γ increases, $\Phi = \tau^2(L + \gamma I_n)^{-1}$ becomes dominated by the term γI_n and the difference between the performance of the proposed method and that of the OLS diminishes.

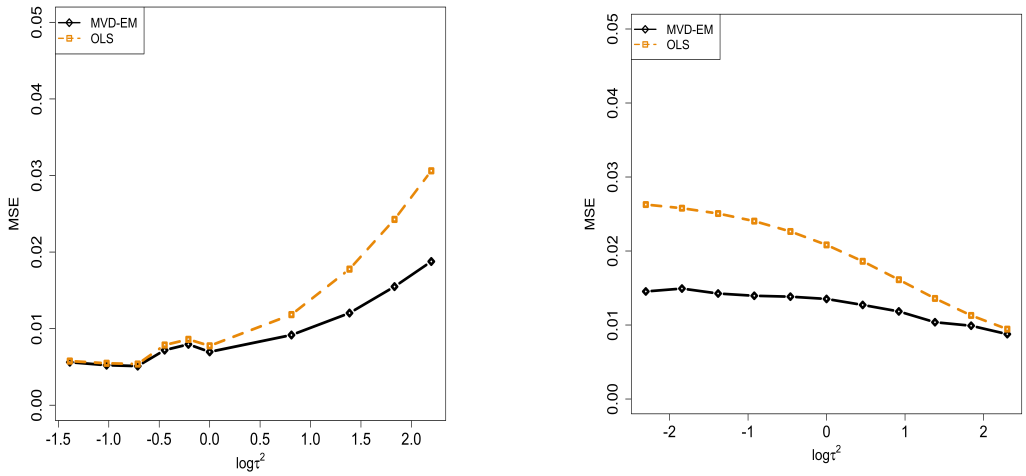


Figure 2.1: In-sample prediction, $n = 200, p = 2$. Left: MSE vs $\log(\tau^2)$, $\gamma = 1$; right: MSE vs $\log(\gamma)$, $\tau^2 = 5$.

Figure 2.2 shows the result of semi-supervised learning in the low dimensional setting. The x-axis is the proportion of observed responses, and the y-axis is the MSE for data points with unobserved Y, i.e.,

$$\frac{\|\hat{Y}_U - \mathbb{E}(Y|X)_U\|^2}{|U|},$$

where U is the set of indices of unobserved responses and $|U|$ is the number of unob-

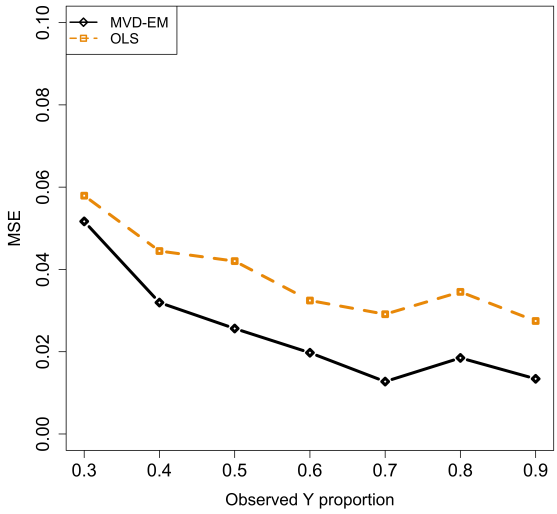


Figure 2.2: Semi-supervised learning, $n = 200, p = 2$. MSE vs the proportion of observed responses, $\tau^2 = 2$, $\gamma = 1$.

served responses. We could see as the proportion of observed responses increases, the prediction error for those unobserved responses gets lower. This is reasonable because we need fully observed rows to estimate $\Sigma_{(p+1)\times(p+1)}$, and more observed rows would render better estimation.

The prediction results in the high dimensional setting are presented in Figure 2.3. As the scale parameter c_Φ increases, the outperformance of MVD based methods also increases. The EM algorithm always achieves a lower MSE than the approximation algorithm. This is because the EM algorithm maximizes the marginal likelihood of Z directly, while the approximation algorithm utilizes the observed data matrix to approximate sample covariances WW^T/p and $\epsilon^T\epsilon/n$, which is less accurate. The scale parameter c_Φ in the high dimensional setting plays a similar role as the parameter τ^2 in the low dimensional setting. They represent the effect of the network and when the scale parameter gets larger, the covariance structure of the matrix variate distribution is dominated by the network effect, and therefore the performance of the estimation procedure based on the i.i.d. assumption degrades.

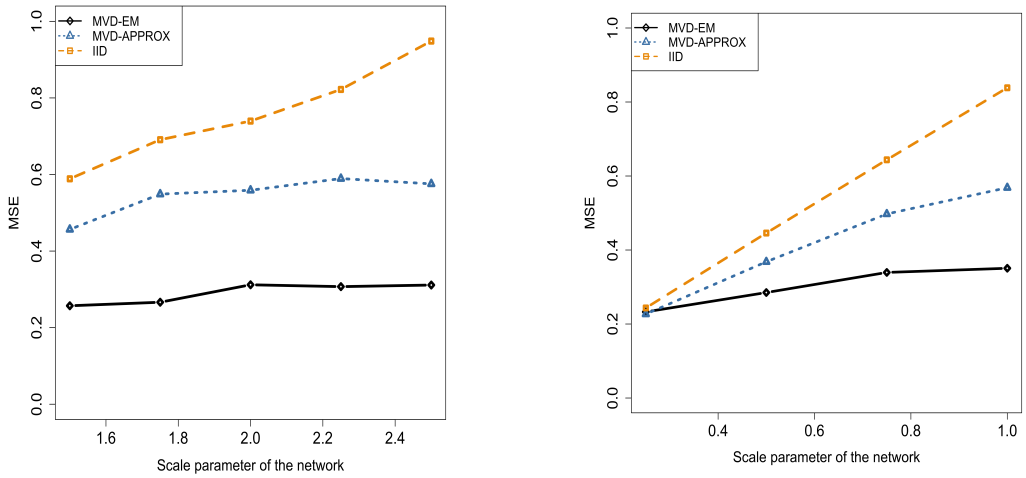


Figure 2.3: In-sample prediction, $n = 50, p = 19$. Left: $\Sigma \sim$ AR(1); right: $\Sigma_{xx} = I_p$.

Tuning hyperparameters Most of the above methods require tuning hyperparameters. We generate a validation set $(X^{\text{val}}, Y^{\text{val}})$ that follows the same distribution as the training set. Then we fit the model on the training dataset with each hyperparameter value, and predict with the fitted model on the validation set, calling it $\widehat{Y}^{\text{val}}$. The parameter that minimizes the prediction error $\|\widehat{Y}^{\text{val}} - Y^{\text{val}}\|^2/n$ on the validation set is selected.

2.4.2 Simulation Studies: Classification

For classification problems, we focus on in-sample prediction in both low dimensional ($n = 100$, $p = 3$) and high dimensional ($n = 50$, $p = 20$) settings.

Parameter settings The covariate matrix and the class labels (X, Y) are obtained by first generating class labels Y by i.i.d multinomial with parameter π, then generating X by (2.10). Here we could generate a mean zero matrix by the same procedure as in Section 2.4.1, then add a mean determined by Y to this data matrix. The number of classes is set to $K = 2$ and $\pi_1 = \pi_2 = 1/2$. For mean parameters of the two classes, we let $\mu_1 = (\mu_0, \mu_0, \ldots, \mu_0) \in \mathbb{R}^p$, and $\mu_2 = (-\mu_0, -\mu_0, \ldots, -\mu_0) \in \mathbb{R}^p$

for some scalar $\mu_0 \in \mathbb{R}$. The mean of the distribution $X|Y$, $M_{n\times p}$, is given by $M_{i\cdot} = \sum_{k=1}^{K} \mathbf{1}(Y_i = k)\mu_k$.

In the low-dimensional ($n = 100$, $p = 3$) setting, the network is generated by a SBM with connection probability matrix

$$\begin{bmatrix} 0.1 & 0.01 \\ 0.01 & 0.1 \end{bmatrix}.$$

Here the community labels are the same as the class labels. Φ is given by $\Phi = \tau^2(L + \gamma I_n)^{-1}$ with $\tau^2 = 10$ and $\gamma = 0.1$. Σ is a fixed positive definite matrix given by

$$\Sigma = \begin{bmatrix} 1 & 0 & 0.5 \\ 0 & 1 & 0.5 \\ 0.5 & 0.5 & 1 \end{bmatrix}.$$

In the high-dimensional ($n = 50$, $p = 20$) setting, the network is generated by a SBM with connection probability matrix

$$\begin{bmatrix} 0.2 & 0.01 \\ 0.01 & 0.2 \end{bmatrix}.$$

Φ is generated in a similar way as in the high-dimensional regression setting, with $tr(\Phi) = 3n$. To generate Σ, we first generate an Erdős-Rényi random graph $\mathcal{G}(p, 0.05)$, then we obtain a covariance matrix Σ following the same procedure as in *Peng et al.* (2009).

We compare the proposed EM-based methods with linear discriminant analysis (LDA). In LDA, it is assumed that given the class label $Y_i = k$, the covariate vector of data point i follows the distribution $X_i|Y_i \sim \mathcal{N}(\mu_k, \Sigma)$ for some Σ, and different data points are independent of each other, i.e., $X_{n\times p}|Y_{n\times 1} \sim \mathcal{MN}(M_{n\times p}, \Sigma \otimes I_n)$.

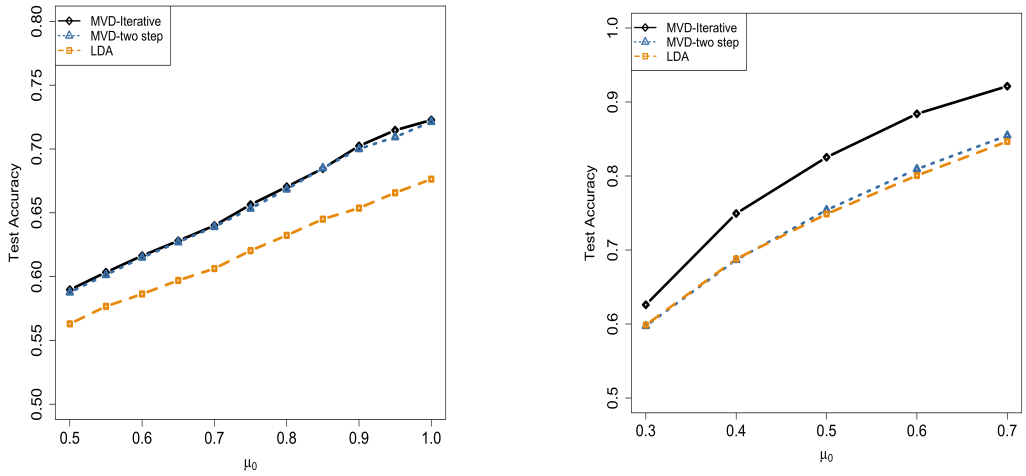


Figure 2.4: Test accuracy vs μ_0. Left: $n = 100$, $p = 3$; right: $n = 50$, $p = 20$.

Results The results for classification are shown in Figure 2.4. As we can see, in the low-dimensional case, the performance of the iterative approach and that of the two-step approach of the proposed estimation methods are similar, and both outperform LDA. However in the high-dimensional setting, the iterative approach for parameter estimation achieves better classification accuracy. Note in the EM algorithm, the updates of Φ and Σ depend on the value of μ, and updating μ by (2.11) also depends on Φ and Σ. The result suggests updating μ within each iteration leads to better parameter estimation as well as classification accuracy.

2.4.3 NASA Central America Grid Data Example

In this subsection, we apply the proposed method to a real-world data example concerning prediction of atmospheric measurements in Central America. Specifically, the dataset contains geographic and atmospheric measures on a coarse 24 by 24 grid covering Central America. There are eight measurement variables: elevation, temperature (surface and air), ozone, air pressure, and cloud cover (low, mid, and high). Except for elevation, all other variables are monthly averages, with observations from Jan 1995 to Dec 2000 (72 months). Thus overall, we have 72 data matrices, with

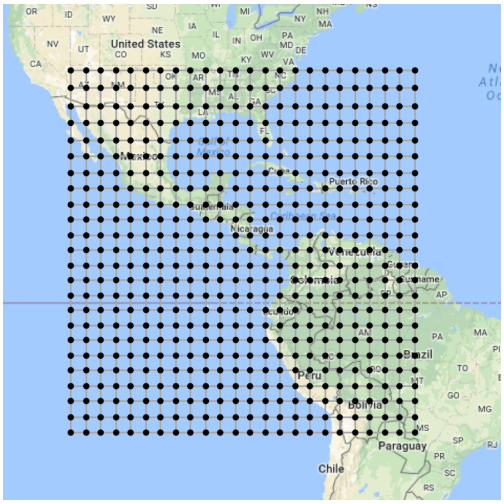


Figure 2.5: Visualization of the grid

each matrix representing a month. Each matrix contains 596 rows (locations) and 8 columns (variables). The network in this data example is a grid network, i.e., each location is a node and its neighbors are the left, right, upper and lower locations adjacent to it (Figure 2.5).

We consider predicting surface temperature here, using other variables as covariates and the grid network information. Temperature variable has seasonal patterns, so we predict the response variable of each month separately. For each month, we have 6 replicates, one from each year. We treat the first four as the training data, the fifth one as the validation set, and test the result on the last year's data matrix.

In the proposed model, we have assumed that the mean of the matrix variate distribution is 0. Therefore when estimating the parameters, we first center each training data matrix by subtracting the sample mean for each column. We denote $\overline{Y}_{\text{train}}$ as the sample mean of the response variable before centering in the training datasets. When predicting the response variable on the testing dataset, we also first center each column of predictors in the testing data. We denote the testing dataset as (X^*, Y^*) and the centered testing data as (X_0^*, Y_0^*). To predict Y^*, we first get $\widehat{Y}_0^*$ by $\widehat{Y}_0^* = \mathbb{E}\left(Y_0^* | X_0^*\right)$ with plug-in estimators, then we use $\widehat{Y}^* = \widehat{Y}_0^* + \overline{Y}_{\text{train}}$ as an

estimator of Y^*. The results are evaluated by the R^2 on the testing dataset, which is defined as $R^2 = 1 - \|Y^* - \widehat{Y}^*\|_2^2/\|Y^* - \overline{Y}^*\|_2^2$. We compare the proposed EM algorithm with the OLS. The results are provided in Table 2.1. We find that the proposed method (MVD) outperforms the OLS in both prediction tasks, suggesting that taking into account the correlation between adjacent geographic locations would improve the prediction accuracy.

Table 2.1: R^2 for predicting the temperature

Month	Jan	Feb	Mar	Apr	May	June
MVD	0.868	0.876	0.861	0.749	0.518	0.721
OLS	0.849	0.851	0.836	0.771	0.504	0.652
Month	July	Aug	Sept	Oct	Nov	Dec
MVD	0.817	0.727	0.823	0.768	0.858	0.885
OLS	0.777	0.678	0.799	0.762	0.858	0.879

2.5 Discussion and Future Work

In this paper, we focus on the problem of predicting a variable of interest using both covariates and network information. We utilize matrix variate models with Kronecker sum covariance structure to model the distribution of node variables. This model is capable of modeling the dependence among data points and among variables. Network information is incorporated to specify the conditional independence between data points that are not linked in the network. Efficient EM algorithm and one step approximation algorithm for parameter estimation have been developed, and theoretical properties of estimators obtained from the latter have been established. We have considered prediction under several settings, including both high-dimensional and low-dimensional data, and in-sample and semi-supervised predictions. The performance of the proposed method performs well in practice, supported by simulation studies and a real-world data example. This framework is flexible and could be used for regression and classification problems, depending on the variable of interest is

continuous or categorical.

The proposed work can be extended in several potential directions. For example, besides the Kronecker sum covariance, we could adapt the regression and classification framework based on matrix variate models with other covariance or precision structures.One possible alternative is modeling the precision matrix as the Kronecker sum of two precision matrix, as proposed in (*Kalaitzis et al.*, 2013):

$$Z_{n\times q} \sim \mathcal{MN}(0, \left(\Sigma^{-1}_{q\times q} \otimes I_n + I_q \otimes \Phi^{-1}_{n\times n}\right)^{-1}).$$

This precision matrix structure specifies the sparse conditional independence among variables of different observations. As in our method, network information could naturally be incorporated in the model, e.g., specifying zero elements in $\Phi^{-1}_{n\times n}$.

A more general but also more challenging extension is to consider mixed matrix variate models that allow modeling the joint distribution of both continuous and discrete variables associated with dependent observations. Mixed graphical models for modeling both continuous and discrete variables when observations are i.i.d. have been considered in *Cheng et al.* (2017); *Fellinghauer et al.* (2013); *Lee and Hastie* (2015). Extending mixed graphical models to mixed matrix variate models is non-trivial since we need to incorporate the correlation among data points as well as among mixed types of variables. Such extensions can not only handle the classification problem where each node is associated with a class label, but are also applicable to regression problems when many covariates are categorical.

CHAPTER III

Joint Latent Space Models for Network Data with High-dimensional Node Variables

3.1 Introduction

Network data that describe the relations or interactions among individuals have been prevalent in many scientific and engineering fields, including but not limited to social media, world wide webs, and neurosciences (*Newman*, 2010; *Kolaczyk and Csárdi*, 2014). In recent years, a collection of statistical models have been proposed to analyze network data appearing in various domains, for example, see *Goldenberg et al.* (2010) for a review. Many of the existing models are based on the assumption that the formation of network links is driven by nodal latent variables. Such models include stochastic block models (*Holland et al.*, 1983), latent space models (*Hoff et al.*, 2002), random dot product graph models (*Young and Scheinerman*, 2007; *Athreya et al.*, 2017), etc. It is critical to estimate the node latent variables accurately, because the estimated latent representations of nodes not only provide insights on the structure of the network, but also can be further used as node features for subsequent tasks, such as node clustering, prediction for node response variables, and network link prediction.

In many real-world networks, the network link information is often collected along

with additional high-dimensional node variables. For example, in an online social network, where nodes represent users and links represent friendship relationships, we also observe users' multiple personal information such as age, gender, and education institution (*Leskovec and Mcauley*, 2012); and in a citation network where nodes represent papers and links represent citation relationships, word frequencies over a large number of words for each paper are recorded as well (*McCallum et al.*, 2000). The dimension of node variables in these applications can be large in the sense that it is comparable to the number of nodes. Existing studies have shown that node variables provide complementary information to network links and often play important roles for estimating the latent structure of the network (*Zhang et al.*, 2016; *Binkiewicz et al.*, 2017; *Newman and Clauset*, 2016). Therefore, it is important to model the network and node variables jointly such that the node variable information can be utilized for improved understanding of the node latent variables in the latent space.

In this paper, we propose a joint latent space model to model network links and high-dimensional node variables simultaneously using shared latent variables. On one hand, as mentioned above, many commonly used network models assume that the network links are determined through node latent variables. Among these models, *latent space models* are probably the most popular one (*Hoff et al.*, 2002) and have been shown to be powerful for capturing many commonly observed features of real-world networks (*Ward and Hoff*, 2007; *Ward et al.*, 2007, 2011; *Friel et al.*, 2016), such as node degree heterogeneity, homophily, and community structures (*Ma and Ma*, 2017). Taking advantage of these nice properties of the network latent space model, we also assume that each node can be represented by a latent vector in a (low) dimensional Euclidean space, and the connecting probability between two nodes depends on the corresponding pair of nodes' positions in the unobserved space. On the other hand, it is also commonly observed that high- or moderate-dimensional variables that are correlated can often be explained in terms of a few unobserved

latent variables as well (*Bai and Li*, 2012; *Wang et al.*, 2017a; *Hair et al.*, 2018). Further, for network data with node variables, the latent variables that explain the observed high-dimensional node variables could be correlated with the latent position variables that explain the network links. This motivates us to consider the joint latent space modeling framework, which uses the shared latent variables to model both parts of the observed information, with the goal of utilizing node variables effectively to help estimate the node latent positions.

Related work Various latent variable based network models have been proposed for modeling network data with node variables, such as the latent space model (*Hoff et al.*, 2002) and its variants (*Hoff*, 2003; *Handcock et al.*, 2007; *Hoff*, 2005, 2008, 2009; *Krivitsky et al.*, 2009; *Sewell and Chen*, 2015, 2016; *Ma and Ma*, 2017). When node covariates are present, existing latent space models usually incorporate such information by including pairwise node variable similarities to model link probabilities (*Hoff et al.*, 2002; *Ma and Ma*, 2017). Such similarity-based approaches have several limitations when node variables are of high dimension. First, majority of existing latent space models adopt Bayesian estimation approaches. The high-dimensional similarity vector would introduce a large number of parameters and additional MCMC sampling, therefore, making the estimation much more computationally challenging. Second, the performance of the similarity-based method would be sensitive to the specific choice of the similarity measure; and it does not model the relationship between the observed node variables and the latent variables. In practice, node variables are often correlated with latent variables and this relationship can be utilized for better understanding of the network structure (*Xu et al.*, 2012; *Yang et al.*, 2013; *Kim and Leskovec*, 2012). Further, from the theoretical perspective, the existing literature has rarely studied the effects of high-dimensional node variables on estimating network latent representations, which is necessary in modern network data analysis with node variables.

Main contributions of the paper. From the modeling perspective, the proposed framework has several advantages in comparison to the existing work. First, we model the relationship between node latent variables and node covariates by a set of shared latent variables, which provides a natural way of borrowing information from node variables to improve the estimation of the latent variables. Second, the proposed model adopts the framework of generalized linear factor models to model the distribution of node variables and, therefore, can handle multiple types of node variables arising in practice (such as continuous, binary, and count variables, etc.). Further, we develop an efficient projected gradient descent algorithm to estimate the model parameters and latent representations, by treating the latent representations as fixed effects. Such an estimation method is computationally more efficient than the Bayesian estimation approaches in the existing literature.

Moreover, from the theoretical perspective, we show that the proposed estimators of the (fixed effect) joint latent space model are error rate optimal. We also establish the corresponding non-asymptotic upper and lower error bounds. In addition, we provide new findings on how the information from both the network and node variables would balance with each other to affect the estimation of latent variables. In particular, we provide a theoretical guarantee that when the dimension of node variables is large enough, borrowing information from node variables would always achieve improvement in estimating node latent positions, in comparison with the results using network link information only. We also investigate how the sparsity level of the network would affect the necessity of including node variables for joint estimation. Our theoretical findings are further supported by extensive simulation studies.

The rest of the paper is organized as follows. In Section 2, we present the joint latent space model and propose the projected gradient descent algorithm that estimates the latent positions using a criterion incorporating both network link information and node variables. In Section 3, we derive theoretical properties of the estimators and

prove that the joint modeling framework which incorporates node variables can improve the estimation accuracy of the latent positions. Further, the estimated latent variables can be utilized for downstream tasks, such as missing value imputation for node variables. We demonstrate the improvements in latent variable estimations and the improvements in associated tasks by simulation studies in Section 4 and a Facebook data example in Section 5. We conclude the paper and discuss about future directions in Section 6.

3.2 Proposed Method

3.2.1 Joint Latent Space Model

Assume we have a network A that is composed of n nodes. For each node i, assume we also observe a vector of covariate variables, denoted by $Y_i \in \mathbb{R}^q$. The matrix of node variables is denoted by $Y = [Y_1, Y_2, \ldots, Y_n]^T \in \mathbb{R}^{n \times q}$. For a general matrix $M \in \mathbb{R}^{m \times n}$, we denote its ith row by $M_{i\cdot}$ and its jth column by $M_{\cdot j}$.

We consider a joint latent space model for modeling network data with high-dimensional node variables. Specifically, we assume each node $i \in \mathcal{V}$ can be represented by a low-dimensional vector $Z_i \in \mathbb{R}^k$ in an unobserved latent space. The latent variables of all the nodes are denoted by $Z = [Z_1, Z_2, \ldots, Z_n] \in \mathbb{R}^{n \times k}$. As commonly assumed in previous network latent space models, two nodes that are close in the latent space are more likely to be connected. Meanwhile, it is also proper to assume that when the two nodes are close in the latent space, they may display similarity regarding the observed traits. For instance, for two individuals who have close latent representations in a social network, they may choose similar jobs or hold similar political perspectives. This naturally leads to the consideration that the latent variables not only model the network connectivity, but also are informative for the node variables. In particular, we make the assumption that the distribution of network links

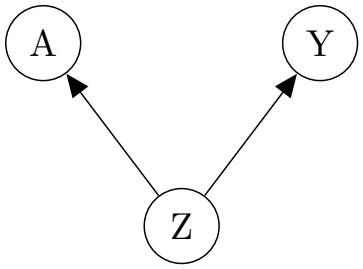


Figure 3.1: Graphical representation of the joint latent space model

and that of node variables are driven by the shared latent variables (Figure 3.1).

For the network A, we assume that for each pair of nodes (i, i'), given their latent positions Z_i and $Z_{i'}$, the presence or absence of an edge between them is determined by the corresponding pair of latent variables and is independent of any other edges. Specifically, for $i < i'$, we assume

$$A_{ii'} = A_{i'i} \overset{ind}{\sim} \text{Bernoulli}(P_{ii'}),$$

with $P_{ii'} = f(Z_i, Z_{i'})$ for some function f. Multiple choices for the function f are available, see *Hoff et al.* (2002), *Hoff* (2003, 2008), and *Ma and Ma* (2017) for examples. In this paper, we consider using the inner-product latent space model (*Hoff*, 2003; *Ma and Ma*, 2017), i.e.,

$$\text{logit}P_{ii'} = \Theta^A_{ii'} = \alpha_i + \alpha_{i'} + Z_i^T Z_{i'}. \tag{3.1}$$

The model specification (3.1) can also be expressed in a matrix form:

$$\text{logit}P = \Theta^A = \alpha 1_n^T + 1_n \alpha^T + ZZ^T,$$

where $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)^T$. We choose the inner-product latent space model to model the network part because of its flexibility to capture commonly observed network characteristics. For example, it allows node degree heterogeneity through the

parameter α_i's, and in general the larger α_i, the more likely that node i connects with other nodes. It also allows for transitivity, i.e., nodes with common neighbors are more likely to connect since their latent positions are more likely to have larger inner product.

Further, we assume that the same latent variables Z are used to describe the multivariate node variables $Y \in \mathbb{R}^{n\times q}$. Specifically, we assume that given Z, the entries in Y are independent and Z models Y through generalized linear models (*Dunn and Smyth*, 2018), with

$$g(\mathbb{E}Y) = \Theta^Y = 1_n\gamma^T + ZB, \tag{3.2}$$

where $\gamma \in \mathbb{R}^q$ and $B \in \mathbb{R}^{k\times q}$ are the "regression" coefficients. For example, when entries in Y are continuous and $g(x) = x$ is the identity mapping, we assume

$$Y_{ij} \overset{ind}{\sim} \mathcal{N}\left((1_n\gamma^T + ZB)_{ij}, \sigma^2\right) \tag{3.3}$$

for some σ^2. When entries in Y are all binary and $g(x) = \log(x/(1-x))$, we have

$$Y_{ij} \overset{ind}{\sim} \text{Bernoulli}\left(\frac{\exp(1_n\gamma^T + ZB)_{ij}}{1+\exp(1_n\gamma^T + ZB)_{ij}}\right). \tag{3.4}$$

More generally, when there are more than one type of variables in Y, we could divide Y into R blocks, i.e., $Y = [Y_1|\ldots|Y_R]$, with each sub-block containing the same type of variables and having its own link function. For the following algorithms and theoretical results, we mainly focus on the case that there is only one type of variables in Y. The corresponding results for Y with multiple variable types can be naturally obtained.

It is worth noting that certain modeling approach for the community detection problem can be considered as having a similar flavor of jointly modeling the dis-

tribution of A and Y via shared latent variables, where the discrete latent variable $Z_i \in \{0,1\}^k$ represents the unobserved community membership of node i. For instance, *Xu et al.* (2012), *Yang et al.* (2013), and *Kim and Leskovec* (2012) assumed that for nodes from the same community or cluster, their edge connections and node variables should follow the common distribution specific to that cluster. In other words, the node latent communities determine the distribution of both A and Y, and therefore information from both A and Y could be used for community detection. Our joint latent space model considers a more general setting where the latent variables could take continuous values in the unobserved latent space.

Note that here we treat Z as fixed effects rather than random. This is due to two reasons. First, our method does not require specific assumptions on the distribution of Z and therefore is more flexible and general; while treating Z as random effects usually needs to make certain parametric assumptions on the distribution of Z. Second, by treating Z as fixed parameters, gradient descent methods can be adopted for parameter estimation and the computation is usually efficient and scalable. On the other hand, when Z is viewed as random effects, the popularly used Bayesian estimation approaches may be computationally expensive.

Remark III.1. Although we assume that models (3.1) and (3.2) share the same set of latent variables Z, this assumption can be easily relaxed. For instance, consider that we have latent variables $Z_A \in \mathbb{R}^{n\times k}$ that model A through (3.1). Meanwhile, there exists another set of latent variables $Z_Y \in \mathbb{R}^{n\times k'}$ that are specifically informative for Y:

$$g(\mathbb{E}Y) = 1_n\gamma^T + Z_Y B. \tag{3.5}$$

If there exists a matrix $W \in \mathbb{R}^{k\times k'}$ such that $Z_Y \approx Z_A W$, then model (3.5) could be rewritten as $g(\mathbb{E}Y) \approx 1_n\gamma^T + Z_A WB = 1_n\gamma^T + Z_A\widetilde{B}$ with $\widetilde{B} = WB$, which gives a good approximation to the model in (3.2). Therefore, even when the two sets of latent variables explaining network and node variables are not exactly the same,

but if there is an approximate linear transformation relationship between them, our proposed joint latent space model is still valid, with Z_A being the shared latent variables that explain both parts of the observed information. In this paper, our main focus is on estimating the latent variables Z that explain the network structure. With additional information from node variables, we are interested in whether the estimation of Z could be further improved.

Identifiability To ensure the joint model to be identifiable, we need to put additional structural constraints on the latent variables Z. First, note that we could add a constant term to both Z_i and Z_j and subtract the corresponding terms from α_i and α_j to keep the distribution of A invariant, so we require that the latent variables are centered, i.e., $JZ = Z$ where $J = I_n - \frac{1}{n}1_n 1_n^T$. This constraint makes Z identifiable up to an orthogonal transformation of its rows. Correspondingly, B is identifiable up to an orthogonal transformation of its columns. Therefore, we further require that the sample covariance of Z, i.e., $Z^T Z/n$, is a diagonal but non-identity matrix. Then the parameters α, Z, B and γ can be uniquely determined.

3.2.2 Estimation

The parameters that need to be estimated are Z, α, B and γ. For the network data A, we consider the loss function as its conditional negative log-likelihood:

$$L_A = -\log P(A|Z,\alpha) = -\sum_{1\le i,i'\le n} \left\{A_{ii'}\Theta^A_{ii'} - f_A(\Theta^A_{ii'})\right\}, \tag{3.6}$$

where $f_A(x) = \log(1+\exp(x))$.

For the node variables Y, we have the negative conditional log-likelihood as

$$L_Y = -\log P(Y|Z,B,\gamma) = -\sum_{\substack{1\le i\le n\\ 1\le j\le q}} \left\{Y_{ij}\Theta^Y_{ij} - f_Y(\Theta^Y_{ij})\right\}, \tag{3.7}$$

where the terms that are irrelevant of Z and γ are omitted. The form of $f_Y(\cdot)$ depends on how the distribution of Y is specified. For example, when Y is continuous as in model (3.3), $f_Y(x) = x^2/2$; and when Y is binary as in model (3.4), then $f_Y(\cdot)$ takes the same form as $f_A(\cdot)$.

We define the objective function as

$$L(Z, \alpha, B, \gamma) = L_A + \lambda L_Y, \tag{3.8}$$

where λ is a weight parameter that controls the information contributed from each part. Our goal is to find the estimators $\widehat{Z}, \widehat{\alpha}, \widehat{B}$ and $\widehat{\gamma}$ such that

$$(\widehat{Z}, \widehat{\alpha}, \widehat{B}, \widehat{\gamma}) = \arg\min_{Z \in \mathbb{R}^{n\times k}, \alpha \in \mathbb{R}^n, B \in \mathbb{R}^{k\times q}, \gamma \in \mathbb{R}^q} L(Z, \alpha, B, \gamma). \tag{3.9}$$

We consider using projected gradient descent algorithm for parameter estimation. At each step, the parameters are updated along the direction that decreases the objective function. In particular, Z and α are updated along the direction of their negative gradients at each iteration. While given all the other parameter, the $\widehat{B}$ and $\widehat{\gamma}$ that minimize the objective function (3.8) can be solved directly. Therefore, we update $\widehat{B}$ and $\widehat{\gamma}$ with those values at each iteration. The algorithm is summarized in Algorithm 3.

Algorithm 3 Projected Gradient Descent Algorithm for Parameter Estimation

1: Input: network adjacency matrix $A \in \mathbb{R}^{n\times n}$; node variables $Y \in \mathbb{R}^{n\times q}$; latent space dimension
2: $k \geq 1$; initial estimates: $Z^0, \alpha^0, B^0, \gamma^0$; step sizes η_Z, η_α; number of iterations T
3: Parameters: Z, α, B, γ
4: For $t = 0, 1, \ldots, T-1$
5: $\quad Z^{t+1} = Z^t - \eta_z \nabla_Z L = Z^t + 2\eta_z (A - f'_A(\Theta^t))(Z^t)^T + \lambda\eta_z \left(Y - f'_Y(Z^t B^t)\right)(B^t)^T$
6: $\quad \alpha^{t+1} = \alpha^t - \eta_\alpha \nabla_\alpha L = \alpha^t + 2\eta_\alpha (A - f'_A(\Theta^t)) 1_n$
7: $\quad (\gamma_j^{t+1}, B_{\cdot j}^{t+1}) = \arg\max_{b \in \mathbb{R}^{k+1}} \sum_i \{Y_{ij}[1, Z_{i\cdot}^t]b - f_Y([1, Z_{i\cdot}^t]b)\}$, $j = 1, \ldots, q$
8: $\quad Z^{t+1} = JZ^{t+1}$
9: Output: $\widehat{Z} = Z^T, \widehat{\alpha} = \alpha^T, \widehat{B} = B^T, \widehat{\gamma} = \gamma^T$

Remark III.2. Algorithm 3 requires initial inputs of several hyperparameters. For the initialization value $Z^0, \alpha^0, B^0, \gamma^0$ of Z, α, B, γ and choice of step size η_Z and η_α, we adapt the initialization method and step size choice proposed in *Ma and Ma* (2017), which proposed to optimize a regularized version of the negative log-likelihood of the network data as in (3.6) to obtain initial estimates Z^0 and α^0. Then we regress Y on Z^0 to get an initialization of B^0 and γ^0. For the step size choice, we let $\eta_Z = \eta/\|Z^0\|_F^2$ and $\eta_\alpha = \eta/(2n)$ for a small and fixed constant η. As for the latent space dimension k, it could be selected through cross validation. As we focus on estimating the latent variables that explain the network links, we consider performing cross validation on the adjacency matrix. In particular, we randomly remove entries from the adjacency matrix, then fit the joint latent space model and predict those missing links based on the fitted values. The process can be repeated for several times and the k that gives the best link prediction performance is chosen from a set of candidate values.

3.3 Theoretical Results

In this section we state our main theoretical results on the estimation of Z under the joint modeling framework. We first show the error bound of the estimators obtained from (3.9). Then we discuss about how the joint modeling framework could improve the estimation of Z.

To study the theoretical properties, we make the following assumptions on parameters.

Assumption III.3. *There exists* $M_1 > 0$ *such that* $-M_1 < \Theta^A_{ii'} < M_1$, *for all* $1 \le i, i' \le n$.

Assumption III.4. *There exists* $M_2 > 0$ *such that* $-M_2 < \Theta^Y_{ij} < M_2$, *for all* $1 \le i \le n, 1 \le j \le q$.

Moreover, we introduce a feasible parameter space as

$$\begin{aligned}\mathcal{F}(Z,\alpha,B,\gamma,M_1,M_2) = \{\Theta \in \mathbb{R}^{n\times(n+q)} \mid \Theta = [\Theta^A,\Theta^Y],\\ \Theta^A = \alpha 1_n^T + 1_n\alpha^T + ZZ^T, \Theta^Y = 1_n\gamma^T + ZB,\\ \max_{1\le i,i'\le n}|\Theta^A_{ii'}| < M_1, \max_{1\le i\le n,1\le j\le q}|\Theta^Y_{ij}| < M_2, JZ = Z\}\end{aligned} \tag{3.10}$$

We denote $\Theta^* = \left[\Theta^{*A},\Theta^{*Y}\right] = [\alpha^* 1_n^T + 1_n(\alpha^*)^T + Z^*(Z^*)^T, 1_n(\gamma^*)^T + Z^*B^*] \in \mathcal{F}$ as the ground truth parameter. Denote $\widehat{\Theta} = \left[\widehat{\Theta}^A,\widehat{\Theta}^Y\right] = \left[\widehat{\alpha}1_n^T + 1_n^T\widehat{\alpha} + \widehat{Z}\widehat{Z}^T, 1_n\widehat{\gamma}^T + \widehat{Z}\widehat{B}\right]$, where $\widehat{Z}$, $\widehat{\alpha}$, $\widehat{B}$ and $\widehat{\gamma}$ are obtained as

$$(\widehat{Z},\widehat{\alpha},\widehat{B},\widehat{\gamma}) = \arg\min_{\Theta\in\mathcal{F}} L(Z,\alpha,B,\gamma).$$

Note that we constrain the true parameters and estimators in the feasible parameter space, mainly for the purpose of theoretical analysis. In practical implementation of Algorithm 3, we do not put additional constraints regarding $\max_{1\le i,i'\le n}|\widehat{\Theta}^A_{ii'}|$ and $\max_{1\le i\le n,1\le j\le q}|\widehat{\Theta}^Y_{ij}|$, and simulation studies indicate that this does not affect the results.

The next theorem provides upper and lower bounds on the estimation error of $\widehat{\Theta}$.

Theorem III.5. *Under Assumptions III.3 and III.4, we have*

$$\frac{1}{\sqrt{n(n+q)}}\mathbb{E}\|\widehat{\Theta}-\Theta^*\|_F \le \frac{\kappa\max(\lambda,1)\sqrt{2k+3}}{\min\left(\min_{|v|<M_1} f_A''(v), \lambda\min_{|v|<M_2} f_Y''(v)\right)}\cdot\frac{1}{\sqrt{n}}, \tag{3.11}$$

where κ is an absolute constant.

Moreover, denote $\overline{\Theta}\in\mathbb{R}^{n\times(n+q)}$ as an arbitrary estimator. When $q=\mathcal{O}(n)$, there exist $\Theta^0\in\mathcal{F}$, $\epsilon_1>0$ and $n_0,q_0>0$ such that for $n>n_0$ and $q>q_0$,

$$P\left(\frac{1}{\sqrt{n(n+q)}}\|\overline{\Theta}-\Theta^0\|_F \ge \frac{\epsilon_1}{\sqrt{n}}\right) \ge \frac{1}{2}. \tag{3.12}$$

The proof is given in Appendix. The result in (3.11) implies that $\|\widehat{\Theta}-\Theta^*\|_F/\sqrt{n(n+q)} = \mathcal{O}_p(1/\sqrt{n})$. Together with the lower bound in (3.12), we can see that the rate of estimation error obtained in Theorem III.5 is optimal. Moreover, the results also indicate that using the network itself, i.e., $q = 0$, we have $\|\widehat{\Theta}^A - \Theta^{*A}\|_F/n = \mathcal{O}_p(1/\sqrt{n})$. Therefore, we achieve the same order of estimation error of parameters under the joint modeling framework, compared to that obtained with the network information only. In particular, the upper bound of $\|\widehat{\Theta}^A - \Theta^{*A}\|_F/n$ is consistent with the results in *Ma and Ma* (2017), which considered the problem of estimating latent variables using network only, by minimizing L_A as defined in (3.6).

While Theorem III.5 indicates that the error bounds of estimators obtained under the joint modeling framework have the same order as that obtained from the network data (without Y), we are also interested in how the additional node variables Y can help the estimation of latent variables. We evaluate the effect of Y in terms of the one-step update analysis. In particular, assuming we have an estimated $\widetilde{Z}$ through the network, for example, from the algorithm proposed in *Ma and Ma* (2017). Suppose with node variables Y, we update $\widetilde{Z}$ for one more step based on Algorithm 3, i.e.,

$$\widehat{Z} = \widetilde{Z} + (1-\widetilde{\lambda})\eta_z(A - f'_A(\widetilde{\Theta}^A))\widetilde{Z} + \widetilde{\lambda}\eta_z(Y - f'_Y(\widetilde{\Theta}^Y))(B^t)^T, \tag{3.13}$$

where $\widetilde{\lambda} = \lambda/(\lambda+2)$. To investigate the properties of $\widehat{Z}$, we make the following additional assumptions.

Assumption III.6. *The dimension of node variables q satisfies the condition that $q = \mathcal{O}(n)$.*

Assumption III.7. *$cov(Z) = Z^TZ/n = diag(\sigma_1, \ldots, \sigma_k) \neq I_k$ is diagonal and the diagonal elements are of constant order $\mathcal{O}(1)$.*

Assumption III.8. *Denote the eigen-decomposition of BB^T/q as $U\Lambda U^T$, where $\Lambda = diag(\lambda_1, \ldots, \lambda_k)$. The eigenvalues are of constant order $\mathcal{O}(1)$.*

Assumption III.6 allows q to grow on a slower or the same order of n. Since $Z \in \mathbb{R}^{n\times k}$ and $B \in \mathbb{R}^{k\times q}$, Assumptions III.7 and III.8 are standard. The following theorem shows that we can achieve more accurate estimation of Z, as long as the dimension of node variables is high enough.

Theorem III.9. *Suppose $\widetilde{Z}$ and $\widetilde{\alpha}$ are estimated from Algorithm 1 in Ma and Ma (2017), and we have a fixed $\widetilde{B}$ satisfying $\|\widetilde{B}-B\|_F^2 = \mathcal{O}(1)$. Then under Assumptions III.3–III.8, there exist positive constants C and $\bar{\lambda}$ such that when $q > Cn$, we have*

$$\mathbb{E}\|\widehat{Z} - Z\|_F^2 < \mathbb{E}\|\widetilde{Z} - Z\|_F^2,$$

for any $\widetilde{\lambda} \in (0, \bar{\lambda})$, where $\widehat{Z}$ is obtained from (3.13).

From Theorem III.9, we see that with additional high-dimensional node variables, we can achieve more accurate estimation of latent variables, in terms of $\mathbb{E}\|\widehat{Z} - Z\|_F^2$. In practice, a $\widetilde{B}$ that satisfies the requirement $\|\widetilde{B} - B\|_F^2 = O(1)$ can be obtained by regressing Y on $\widetilde{Z}$.

Theorem III.9 relies on specific $\widetilde{Z}$ and $\widetilde{\alpha}$. More generally, we consider the scenario where we are given initial estimates of Z, α, B, γ, denoted by $\widetilde{Z}$, $\widetilde{\alpha}$, $\widetilde{B}$, $\widetilde{\gamma}$, respectively, satisfying the conditions that $\|\widetilde{Z}-Z\|_F^2 = \mathcal{O}(1)$, $\|\widetilde{\alpha}1_n^T - \alpha 1_n^T\|_F^2 = \mathcal{O}(n)$, $\|\widetilde{B}-B\|_F^2 = \mathcal{O}(1)$, and $\|\widetilde{\gamma} - \gamma\|_2^2 = \mathcal{O}(1)$. The following proposition provides implications on under what scenarios the joint modeling framework can help better estimate the latent variables Z.

Proposition III.10. *Given $\widetilde{Z}$, $\widetilde{\alpha}$, $\widetilde{B}$, $\widetilde{\gamma}$ that satisfy the above required conditions, we consider to update $\widetilde{Z}$ one step further by* (3.13) *and obtain a $\widehat{Z}$. Under Assumptions III.3–III.8, there exists an optimal $\widetilde{\lambda}_{opt}$ such that $\mathbb{E}\|\widehat{Z} - Z\|_F^2$ is minimized, and $\widetilde{\lambda}_{opt}$ is given by* (3.14). *Under a proper choice of $\widetilde{\lambda}$, the joint modeling framework can achieve a mean square error of $\widehat{Z}$ that is at least as good as the results when using information from A or Y only.*

The proof of Theorem III.9 and Proposition III.10 are given Appendix. Taking the case when Y is continuous as an example, by calculation we can obtain a more explicit expression of $\widetilde{\lambda}_{opt}$, which provides some insights on how the information from both parts balance with each other. As shown in the Appendix, we have $\widetilde{\lambda}_{opt}$ as

$$\widetilde{\lambda}_{opt} = \frac{\widetilde{T}_A - \widetilde{T}_{AY} + \widetilde{e}_A}{\widetilde{T}_Y + \widetilde{T}_A - 2\widetilde{T}_{AT} + \widetilde{e}_A + \widetilde{e}_Y}, \tag{3.14}$$

where the terms $\widetilde{T}_Y$, $\widetilde{T}_A$ and $\widetilde{T}_{AY}$ are defined in Appendix, and

$$\widetilde{e}_Y = \rho_2^2 nq^{-2}\sigma^2 tr(\widetilde{B}\widetilde{B}^T),\ \widetilde{e}_A = \rho_1^2 n^{-2} tr((I_n \otimes \widetilde{Z})^T diag(vec(\sigma'(\Theta)))(I_n \otimes \widetilde{Z})),$$

for some constants ρ_1 and ρ_2.

Proposition III.10 and the expression of $\widetilde{\lambda}_{opt}$ have the following implications. First, note that a positive $\widetilde{\lambda}_{opt}$ suggests incorporating information from node variables is preferred. We can show that the denominator of $\widetilde{\lambda}_{opt}$ is always positive, then a positive numerator or equivalently a large $\widetilde{T}_A - \widetilde{T}_{AY}$ would lead to such a case. By the calculation in the Appendix, we can see that an overall sparser network would more likely lead to a larger $\widetilde{T}_A - \widetilde{T}_{AY}$. Therefore, when the information from the network part is relatively limited, borrowing information from node variables would be preferred and helpful. Second, when the numerator is positive, a smaller denominator would lead to a larger $\widetilde{\lambda}_{opt}$. In particular, when controlling all the other parameters and increasing q, the term $\widetilde{e}_Y$ would become smaller. This suggests that when node variables are of higher dimension or contain richer information, more weight should be put on the node variables part to improve the estimation of Z. Finally, suppose we do not choose the optimal $\widetilde{\lambda}$ but fix it in the one step estimation. In Appendix we also calculates the difference between taking a non-zero $\widetilde{\lambda}$ and $\widetilde{\lambda} = 0$, and the result also suggests that when the dimension of node variables increases or the network gets sparser, the effect of incorporating node variables in terms of estimating Z becomes

more significant. In Section 3.4, we demonstrate how network information and node variables information balance with each other, especially the relationship between the dimension of node variables and the optimal weight as well as the influence of the dimension of node variables dimension and the network density.

3.4 Simulation Studies

3.4.1 Effect of the Dimension of Node Variables

To study how information borrowed from Y can affect estimating latent variables, we compare the estimation of Z using the network latent space model and the joint latent space model. For network latent space model, we consider the version without covariates to demonstrate how node variables can be useful in improving the estimation of Z.

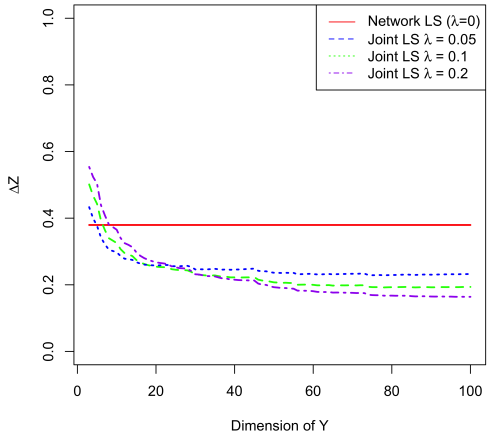
We first study the effect of node variable dimension. We set $n = 200$ and $k = 2$ or 4. We vary q from 2 to 100 to study how the dimension of node variables affects the estimation of Z. The model parameters are specified as follows:

- Generate the degree heterogeneity parameters $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n)$, with each element i.i.d from $U[-0.25, -0.75]$;
- Generate k latent vector centers $\mu_1, \ldots, \mu_k \in \mathbb{R}^k$ with coordinates i.i.d. from $U[-1, 1]$;
- Generate latent variables $Z \in \mathbb{R}^{n \times k}$: first generate a matrix $Z_0 \in \mathbb{R}^{n \times k}$ such that each entry is i.i.d. $\mathcal{N}(0, 1)$. Then we divide n data points equally into k subsets, and for points in each subset, add $\mu_1, \ldots, \mu_k$ to them respectively. Lastly we transform Z by 1) setting $Z = JZ$, 2) normalizing Z such that $\|ZZ^T\|_F = n$, and 3) rotating $Z = ZR$ for some rotation matrix R such that the covariance of Z is a diagonal matrix;

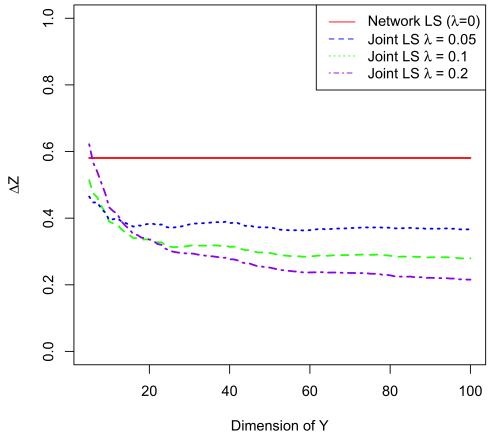
- Generate the coefficients $B \in \mathbb{R}^{k\times q}$, with each entry i.i.d from $\mathcal{N}(0,1)$.

After setting the parameters, we generate A based on model (3.1) and generate Y based on either model (3.3) or model (3.4). Under the considered parameter settings, the average density of the networks is about 0.30. Each setting is repeated 30 times. For each replication, we fit both the network inner-product latent space model and the joint latent space model to obtain estimations of Z, denoted by Z_{net} and Z_{joint} respectively. We evaluate the performance of each method using the criterion $\Delta_Z = \|\widehat{Z}\widehat{Z}^T - ZZ^T\|_F^2/\|ZZ^T\|_F^2$. Figure 3.2 shows the average results of the 30 replications, and we can see that as the dimension of Y increases, the estimation of Z improves, and the joint latent space model starts to outperform the network latent space model even when q is relatively small, indicating the constant C in Theorem III.9 is of small value. Figure 3.2 also demonstrates that overall the improvement of the joint latent space model over the network latent space model is robust to the choice of λ, though the specific value of λ may affect how much improvement we could obtain by incorporating node variables.

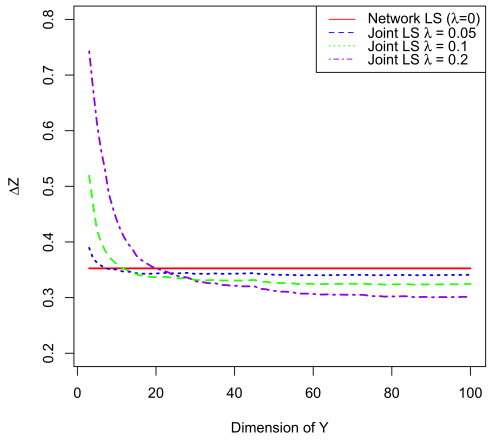
Optimal λ versus dimension of node variables. Recall λ is the parameter that balances the weights of information contributed from the network part and the node variables part. Intuitively, when the signal from A is relatively weak and the information from Y is relatively rich, a larger λ would be preferred. In practice λ could be selected by cross validation. Here our main goal is not hyperparameter tuning, but to investigate how the optimal λ would change as the dimension of Y changes. The optimal λ refers to the λ value which returns the Z_{joint} that minimizes Δ_Z. For each q, $2 \le q \le 100$, we fitted the joint latent space model by varying λ from a set of possible values (ranging from 0.01 to 0.5). Each λ would give a corresponding estimated $\widehat{Z}$ and the $\widehat{\lambda}$ that gives the smallest Δ_Z is selected. Figure 3.3 shows the optimal $\widehat{\lambda}$ versus the dimension of Y. As expected, when q increases, the optimal $\widehat{\lambda}$ increases, indicating that more weights are put on the information from Y to obtain



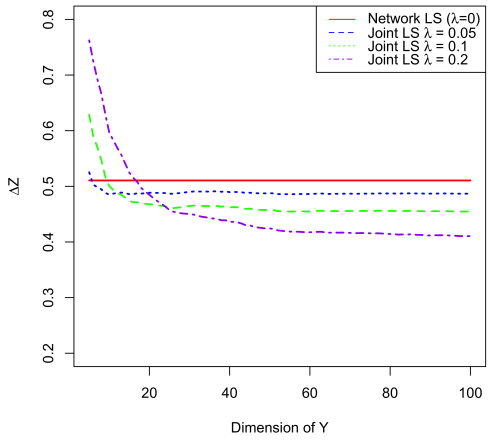
(a) $n = 200, k = 2$, continuous Y



(b) $n = 200, k = 4$, continuous Y



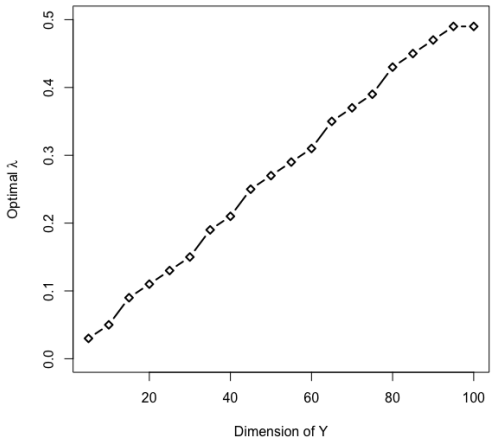
(c) $n = 200, k = 2$, binary Y



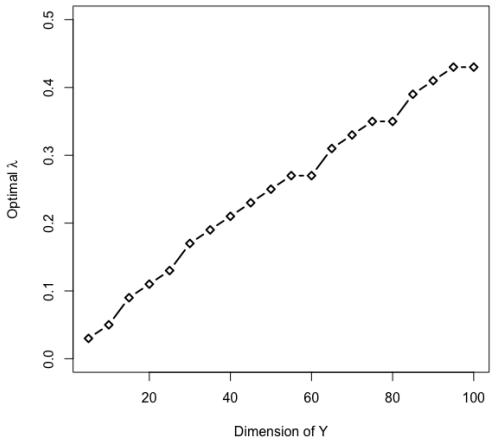
(d) $n = 200, k = 4$, binary Y

Figure 3.2: Estimation of Z versus Dimension of Y

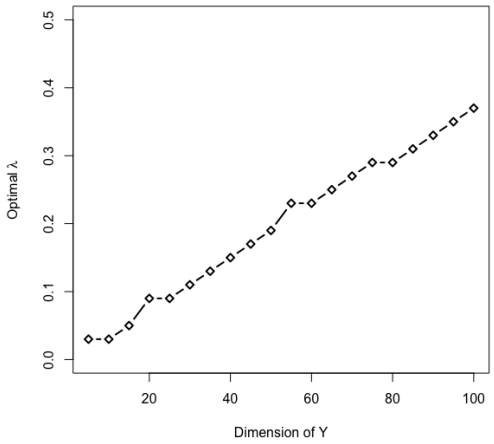
a better estimation of latent variables.



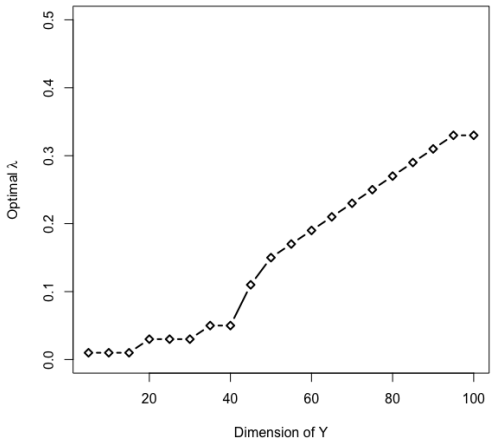
(a) $n = 200, k = 2$, continuous Y



(b) $n = 200, k = 4$, continuous Y



(c) $n = 200, k = 2$, binary Y



(d) $n = 200, k = 4$, binary Y

Figure 3.3: Optimal λ versus Dimension of Y

3.4.2 Effect of Network Density

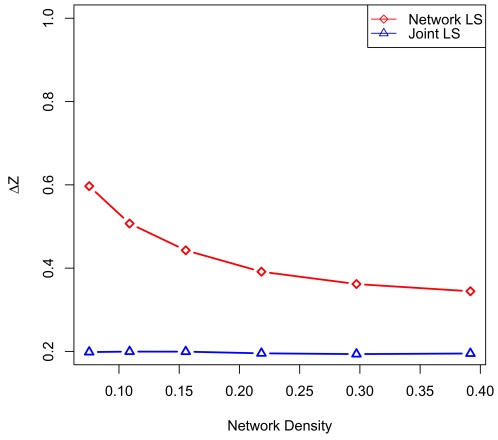
Section 3.4.1 shows that the more information provided from Y, the more improvement we could obtain in the estimation of Z. As a counterpart, in this subsection we demonstrate that when the information from Y is fixed, how does the density of the network affect the difference in the performances of the two models. We fixed $q = 100$

and change the parameter settings in the network. The density of the network is controlled by varying the range of the node degree heterogeneity parameters α, specifically varying from $\alpha_1, \ldots, \alpha_n \sim U[-0.375, -0.125]$ to $\alpha_1, \ldots, \alpha_n \sim U[-2.25, -0.75]$. Z and B are set in the same manner as in Section 3.4.1. Now under different settings of α, the network density ranges from 0.08 to 0.39. We again repeat the simulation 30 times under each setting.

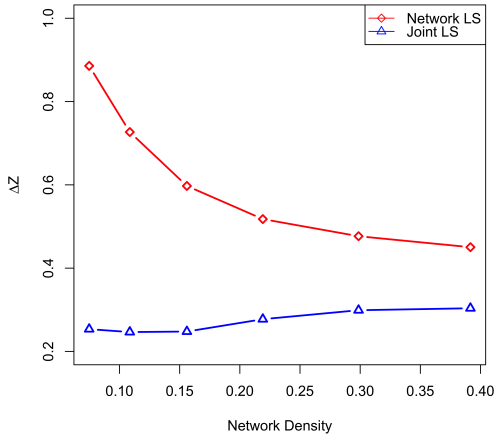
Figure 3.4 shows the estimation of Z based on Z_{net} and Z_{joint} under different levels of the network density. As the network gets sparser, the result of Z estimation using A only gets worse, while the performance of Z_{joint} is relatively stable. This especially suggests the necessity of incorporating node variables in estimating Z, when the network is relatively sparse and may not provide enough information.

3.4.3 Community Membership Estimation

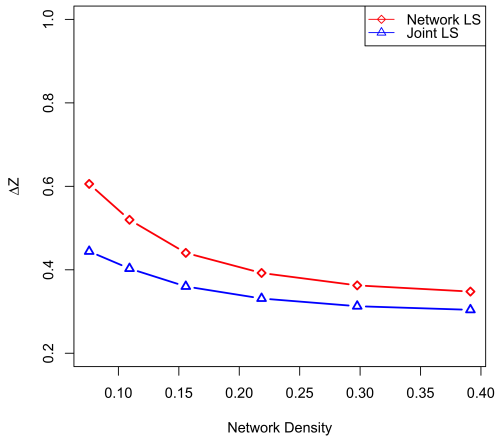
In this subsection we demonstrate the improvement in estimation of latent variables in a more intuitive way by considering a special case of Z. We generate $\alpha_1, \alpha_2, \ldots, \alpha_n \overset{iid}{\sim} U[-3, -1]$, therefore the networks are relatively sparse. For $k = 2$, we set the first $\lfloor n/2 \rfloor$ rows of Z as $(1, 0)$ and the rest $\lfloor n/2 \rfloor$ rows of Z as $(0, 1)$. In other words, the latent variables Z now represent node community memberships. For $k = 4$, we set Z in a similar manner such that there are $\lfloor n/k \rfloor$ data points in each community. The other parameters are specified in the same way as in Section 3.4.1. We estimate Z without relying on the specific structure of Z, and fit the network latent space model and the joint latent space model respectively. The estimated Z_{net} and Z_{joint} from one realization of A and Y are shown in Figure 3.5. We can see that under the setting where the network itself may not provide enough information about the latent variables, using the network information only could not separate all k communities well. However, with additional information from the node variables, we obtain better node clustering.



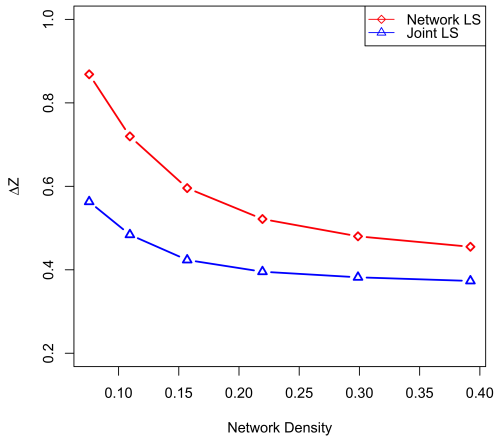
(a) $n = 200, k = 2$, continuous Y



(b) $n = 200, k = 4$, continuous Y

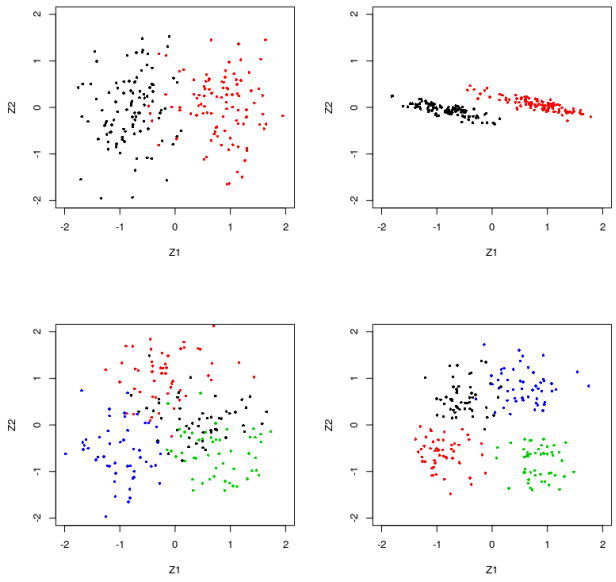


(c) $n = 200, k = 2$, binary Y

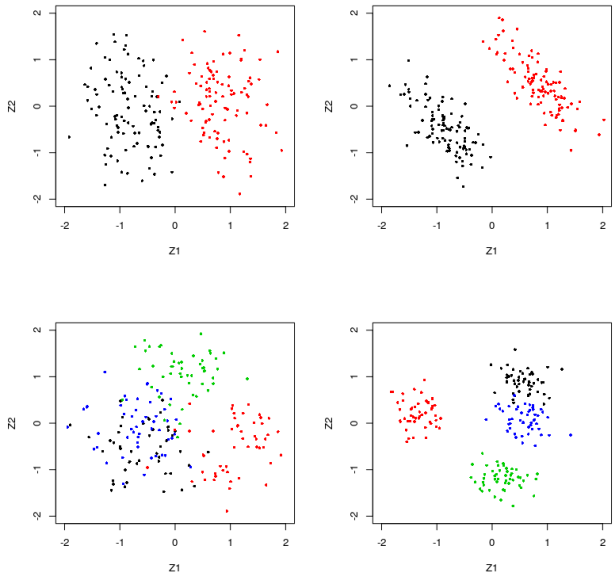


(d) $n = 200, k = 4$, binary Y

Figure 3.4: Estimation of Z versus Network Density



(a) $n = 200, k = 2$ (upper) or 4 (lower), Z_{net} (left) and Z_{joint} (right), continuous Y



(b) $n = 200, k = 2$ (upper) or 4 (lower), Z_{net} (left) and Z_{joint} (right), binary Y

Figure 3.5: Estimation of Z for Community Membership

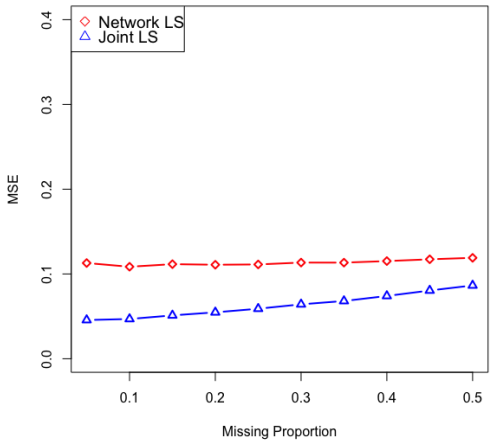
3.4.4 Node Variable Missing Value Imputation

In many applications, estimating node latent positions is an initial step, and such estimation can be further used for downstream tasks. For example, when node variables contain missing values, we could use the estimated $\widehat{Z}$ and the fittted $\widehat{B}$ to impute those missing entries. In this subsection, we demonstrate how the improvement in Z estimation could help with such downstream tasks.

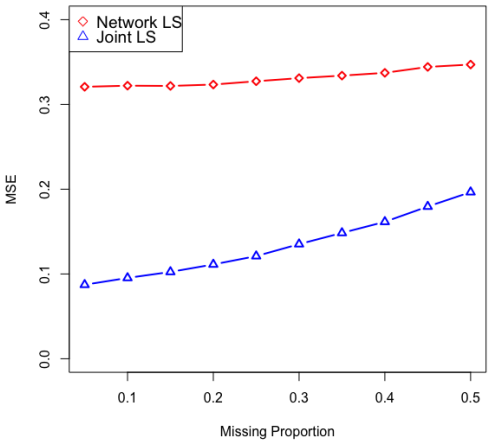
We generate A and Y based on the parameter setting in Section 3.4.1 with $n = 200$, $k = 2$ or 4, and $q = 100$. For each of the replications, we randomly select a fixed proportion of entries from Y and set them as missing. the latent positions Z are estimated using the network latent space model and the joint latent space model respectively. When fitting the joint latent space model, we only utilize the entries in Y that are observed. After obtaining the estimated $\widehat{Z}$, for the jth column of Y, $1 \leq j \leq q$, we regress those observed Y_j's on $\widehat{Z}_{obs}$ to obtain an estimated $\widehat{B}_j$, then we predict those missing entries in Y_j's by $\widehat{Z}_{miss}\widehat{B}_j$. Here the subscripts denote the indexes of observed and missing entries of Y_j respectively. We evaluate the performance of prediction by the mean square error for continuous Y and the AUROC for binary Y, based on all missing entries in Y. The results are shown in Figure 3.6, and we can see that the joint latent space model provides much better missing value imputation than the network latent space model. Further, as the missing proportion increases, since the information from Y becomes limited, the advantage of the joint latent space model becomes less prominent.

3.5 Real Data Example

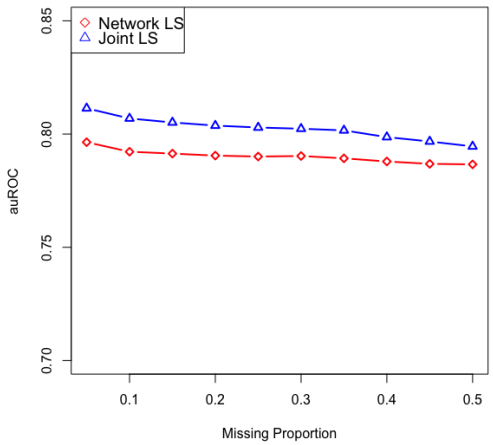
In this section, we demonstrate the proposed model on a Facebook social circle data for the task of node variable missing value imputation. The dataset consists of 10 different networks, each representing an ego network of a selected user, where the



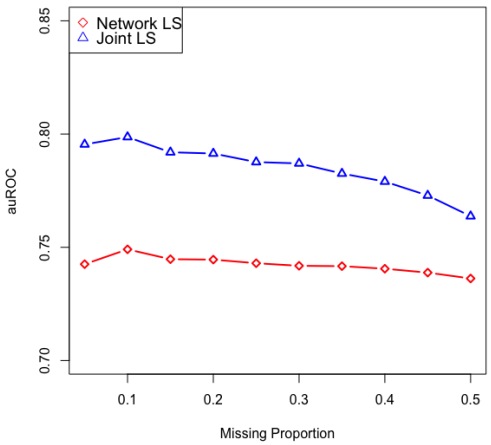
(a) $n = 200, k = 2$, continuous Y



(b) $n = 200, k = 4$, continuous Y

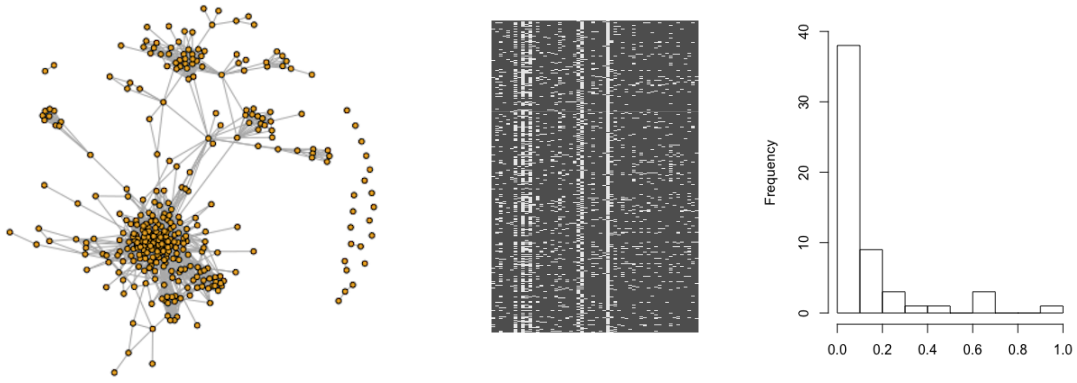


(c) $n = 200, k = 2$, binary Y



(d) $n = 200, k = 4$, binary Y

Figure 3.6: Missing Value Imputation Results



(a) (b) (c)

Figure 3.7: Example of an ego-network: (a) Circle 1 network; (b) node variable matrix; (c) number of variables vs mean of the variable.

ego network is defined as the network between all the user's friends. See Figure 3.7 for an example of an ego network. For each ego network, the data also records the users' anonymized variables. For example, the original dataset contains a variable 'political = Democratic Party', and the curated dataset would transform this into 'political = anonymized feature 1'. The variables associated with each node in this data example are all binary. We analyzed 8 out these 10 networks as two of the networks have relatively few nodes and several variable columns associated with them contain only one or a few 1s. For each network, we also remove the variables that have too many or too few 1s in that variable. If a variable of a user is missing, it is of interest to impute the missing value based on the user's own profile as well as his/her social connections.

We randomly sample 5% or 10% of the entries in the node variable matrix Y and set them as missing. Then we fit the network latent space model and the joint latent space model respectively to obtain estimated $\widehat{Z}_{net}$ and $\widehat{Z}_{joint}$. After obtaining the estimated $\widehat{Z}$, we make predictions on those missing entries following the same procedure as described in Section 3.4.4.

	n	q	density	5% missing AUC (net) AUC (joint)	10% missing AUC (net) AUC (joint)
circle 1	347	56	0.042	0.840 0.884	0.841 0.880
circle 2	755	66	0.105	0.865 0.873	0.861 0.880
circle 3	792	100	0.045	0.856 0.901	0.857 0.894
circle 4	1045	153	0.049	0.867 0.871	0.865 0.870
circle 5	547	58	0.03	0.852 0.905	0.848 0.901
circle 6	227	87	0.124	0.832 0.850	0.831 0.851
circle 7	159	55	0.134	0.809 0.836	0.806 0.837
circle 8	170	37	0.115	0.812 0.854	0.815 0.841

Table 3.1: Node variable missing value imputation results for Facebook data. Upper: AUC obtained by $\widehat{Z}_{net}$; Lower: AUC obtained by $\widehat{Z}_{joint}$.

The average AUROC over 30 replications are summarized in Table 3.1, where each cell records the AUROC obtained based on $\widehat{Z}_{net}$ and $\widehat{Z}_{joint}$ respectively. The results indicate that the joint latent space model consistently achieves better performance than the network latent space model across all 8 networks. Note that the AUROC obtained by fitting the network latent space model is already promising, which suggests that the network itself contains substantial information. However, we can still achieve noticeable improvement after incorporating the node variable information. This implies that the latent position variables are associated with the node variables and the joint estimation can help achieve more accurate estimation and imputation results.

3.6 Conclusion and Discussion

In this paper, we propose a joint latent space model that extends the network latent space model by assuming the shared latent positions Z not only explain the network information but also explain the node variables. We assume the latent variables are related to the network adjacency matrix through the inner-product latent space model, and are related to the multivariate node variables through generalized linear models. The latent positions Z are viewed as fixed values and are estimated through a projected gradient descent algorithm. We investigated whether incorporating high-dimensional node variables into modeling and borrowing information from there to estimate latent positions could improve the estimation, in comparison to using network information only. The results are supported by theory and simulation studies. Further, we consider common tasks on networks, such as node variable missing value imputation, and show that the improvement in latent variables estimation could further help the performance of such tasks as well.

In our simulation and real data examples, we use all n data points with observed information for model fitting and we obtain estimated model parameters $\widehat{Z} \in \mathbb{R}^{n\times k}$, $\widehat{\alpha} \in \mathbb{R}^n$, $\widehat{B} \in \mathbb{R}^{q\times k}$ and $\widehat{\gamma} \in \mathbb{R}^q$. The prediction for those missing node variables are also performed on these n data points. One possible extension is to consider an inductive setting where we make prediction on a new node, which is not present during the training stage, with partial observed information. For example, consider the cold-start scenario where only the new node's variables Y_{n+1} are observed but its link information to all the other n nodes are unknown. This is a case commonly seen in social networks, where newly registered users only provide their personal information but have not connect with any other users. To predict links between the $(n+1)$th node and the previous n nodes, we can first estimate $\widehat{Z}_{n+1}$ by regressing Y_{n+1} on the fitted $\widehat{B}$, then compare $\widehat{\alpha}_i + \widehat{Z}_i^T\widehat{Z}_{n+1}$, for $i = 1, \ldots, n$, to rank which nodes have higher probabilities to connect with the new node. Correspondingly, we

can also make prediction on the $(n+1)$th node variables, when we only know its link information with other nodes.

It is also worth noting that in this paper, we make the assumption that the network and node variables share the same set of latent variables. However, it may be more practical to assume that while they share a set of latent variables, there may exist some other latent variables distinct to explain each part. Specifically, we assume that $Z_1 \in \mathbb{R}^{n\times k_1}$ are latent variables that are uniquely informative for network A, and $Z_A = [Z_1, Z] \in \mathbb{R}^{n\times(k+k_1)}$. Then the network follows the latent space model with $\mathbb{E}A = P$, where

$$\text{logit}P = \Theta^A = \alpha 1_n^T + 1_n \alpha^T + Z_A Z_A^T.$$

Correspondingly, we assume that there are latent variables $Z_2 \in \mathbb{R}^{n\times k_2}$ uniquely explaining Y, and $Z_Y = [Z_2, Z] \in \mathbb{R}^{n\times(k+k_2)}$. Then the node variables Y follow the generalized linear models with

$$g(EY) = Z_Y B,$$

where $B \in \mathbb{R}^{(k+k_1)\times q}$. It can be shown that the parameters Z_A and Z_Y can be jointly estimated, but it is beyond the scope of the current paper, and we leave the investigation for future work.

CHAPTER IV

A Flexible Latent Space Model for Multilayer Networks

4.1 Introduction

In many applications, individuals often interact with each other through more than one type of relations. For example, people can be coworkers or friends (*Lazega et al.*, 2001); or interactions among individuals can happen through social activities or money exchanges (*Banerjee et al.*, 2013). Multiple types of relationships among entities naturally introduce multilayer networks, where different networks share matched node set, while each single network has a distinct edge type defined through a type of relationship. Tools designed for a single network can be naively used to deal with multilayer networks in two ways: either aggregating multiple layers into a single network or analyzing each single network separately. However, aggregating multilayer networks may lose the specific information contained in each layer, while analyzing each network separately does not leverage the information that may be shared across different relations. Therefore, it is of importance to design tailored tools for multilayer network data.

Real-world multilayer networks are often observed with both homogeneity shared between different layers and heterogeneity retained within each layer. For example,

nodes usually have their own intrinsic traits that are consistent across different relations, and at the same time, specific activity levels of individual nodes and the overall network connecting characteristics, such as the edge density or homophily patterns, may vary across different layers. Figure 4.3 provides an example on multiple social networks among the same set of people and demonstrates the heterogeneous node individual behaviors in different social relations. In this paper, we propose a flexible model for multilayer networks which uses the latent space model for a single network *Hoff et al.* (2002) as building blocks, with the goal of capturing aforementioned observed characteristics for multilayer networks. Specifically, we assume each node is represented by a common latent vector shared across layers such that the commonality among layers is kept. Moreover, we assume within each layer, nodes have layer-specific individual effects and connecting patterns, accommodating distinctions between layers. Model specifications and a scalable model fitting algorithm are introduced in Section 4.3. In Section 4.4, we establish theoretical properties of the maximum likelihood estimators of the proposed model. In particular, we study the relationship between the estimation of nodes' common latent representations and the number of layers. The theoretical properties are further supported by simulation studies in Section 4.5. We also demonstrate the performance of the proposed model in terms of latent variables estimation and link prediction on real-world examples in Section 4.6.

The main contributions of this paper include two aspects. The first one is on the model specification. Our proposed model is more flexible in comparison to existing models in the literature (summarized in Section 4.2) in the sense that it allows layer-specific node individual effects, and such flexibility is shown to be necessary when fitting real-world multilayer networks. Introducing these additional layer-specific node individual parameters brings non-trivial challenges for studying theoretical properties of the model, because the mean structure of the resulting multilayer networks

will go beyond the low-rank assumption. The second contribution of our work is on theory, in which we prove that the estimation error of the layer-shared node representations is inversely proportional to the number of layers. This result provides the insight that leveraging multilayer networks for joint estimation is more beneficial than separate estimation with single networks. To the best of our knowledge, this is the first theoretical guarantee on latent variable estimation for multilayer latent space models. Moreover, since our model contains several tensor factorization models (e.g., *Nickel et al.* (2011); *Nickel and Tresp* (2013)) as special cases, our results also provide theoretical support for these models, which is less studied in the literature.

4.2 Related Work

In recent years there has been a growth of statistical models for multilayer networks. The majority of the work extends the tools for modeling a single network to jointly modeling multiple networks, and examples include but are not limited to extensions of: the stochastic block model (SBM) (*Han et al.*, 2015; *Paul and Chen*, 2015, 2020), the mixed membership SBM (*De Bacco et al.*, 2017), the random dot-product graph model (*Levin et al.*, 2017; *Wang et al.*, 2017c; *Nielsen and Witten*, 2018; *Arroyo et al.*, 2019), and the latent space model (*Gollini and Murphy*, 2016; *Salter-Townshend and McCormick*, 2017; *D'Angelo et al.*, 2018; *DAngelo et al.*, 2019). We chose to build on the latent space model due to its flexibility in capturing commonly observed network characteristics, such as node degree heterogeneity, transitivity, homophily, etc.

Latent space models for a single network are first proposed in *Hoff et al.* (2002), and their variants are further developed in *Hoff* (2003, 2008) and *Ma and Ma* (2017). *Gollini and Murphy* (2016) and *DAngelo et al.* (2019) proposed latent space models for multilayer networks, with the assumption that the latent representations for each node are the same across all layers and the variation between networks is captured

through layer-specific parameters that control overall network characteristics, such as edge density or homophily patterns. The assumption of common node representations implicitly suggests that a node has consistent behaviors through all layers and the nodes that behave similarly in one layer should also behave similarly in other layers. This assumption is relatively strict, as it does not reflect the node-level differences across different layers. *Salter-Townshend and McCormick* (2017) allows each node to have a distinct latent representation in each layer. However, due to their specific model assumption, these latent representations are "conditional" and therefore not straightforward to interpret. *D'Angelo et al.* (2018) extends *Gollini and Murphy* (2016) and incorporates layer-specific node effects in each layer. This work is in spirit the closest to our proposed model. However, it considers a different family of latent space models and adopts Bayesian estimation for model fitting, which is computationally much more expensive, and further, there is no theoretical guarantee on model estimation.

Multilayer networks sometimes also refer to dynamic or time-evolving networks (*Sewell and Chen*, 2015, 2017; *Gupta et al.*, 2018), in which connections among the same set of nodes are recorded at different timestamps. The focus is often on the dependency between different layers due to the time order, so the modeling framework is different from that of multiple types of relations. Lastly, multilayer networks can be viewed as a three-way tensor, where the first two dimensions are along the nodes and the third dimension is along the layers. Tensor factorization methods *Tucker* (1966); *De Lathauwer et al.* (2000); *Nickel et al.* (2011); *Nickel and Tresp* (2013) have often been utilized for analyzing multi-relational data. Some special cases of our model reduce to existing tensor factorization models, such as the logistic RESCAL model *Nickel and Tresp* (2013). Therefore, the estimation approach and theoretical results we develop can be directly applied in these cases.

4.3 Proposed Model

Motivated by phenomena observed in real-world multilayer networks and limitations in existing work, we aim to propose a model that has the following properties. First, it should be able to capture the homogeneity and heterogeneity across multiple layers simultaneously. In particular, it should allow node individual effects to vary between layers. Secondly, model parameters should be straightforward to interpret. Lastly, scalable estimation approaches can be developed and theoretical guarantees can be established.

We start with introducing notations. Assuming the multilayer networks are composed of R different networks over a common set of n nodes, with each network representing one type of relation. For $r = 1, \ldots, R$, the rth layer network is represented by a binary adjacency matrix $A^{(r)} \in \{0,1\}^{n \times n}$, where $A_{ij}^{(r)} = A_{ji}^{(r)} = 1$ if node i and node j are connected in the rth relation and $A_{ij}^{(r)} = 0$ otherwise. Stacking the R adjacency matrices together, we obtain a three-way adjacency tensor, denoted by $A = [A^{(1)}; \ldots; A^{(R)}] \in \{0,1\}^{n \times n \times R}$. We focus on binary entries in the adjacency tensor in this paper, though the model can be naturally extended to multilayer networks with more general types of entries (continuous, count, etc.).

4.3.1 Latent Space Model for Multilayer Networks

We extend the main idea in the latent space model for a single layer network, where the connecting probability between two nodes depends on their latent representations in an unobserved Euclidean space. Specifically, we assume that each node i is represented by a unique latent vector $U_i \in \mathbb{R}^k$. Given node latent vectors and layer-specific parameters, we assume connectivity between each pair of nodes i and j in all layers are conditionally independent Bernoulli random variables, i.e.,

$$A_{ij}^{(r)} \overset{ind}{\sim} \text{Bernoulli}\left(P_{ij}^{(r)}\right),$$

where

$$\text{logit}\left(P_{ij}^{(r)}\right) \coloneqq \Theta_{ij}^{(r)} = \alpha_i^{(r)} + \alpha_j^{(r)} + U_i^\top \Lambda^{(r)} U_j, \tag{4.1}$$

for $i, j = 1, \ldots, n$ and $r = 1, \ldots, R$. Note $\alpha^{(r)} = (\alpha_1^{(r)}, \ldots, \alpha_n^{(r)})^\top \in \mathbb{R}^n$ are node degree heterogeneity parameters for layer r. Specifically, when all other parameters are fixed, the larger $\alpha_i^{(r)}$, the more likely that node i connects with other nodes in the rth layer. The $\alpha^{(r)}$s are distinct across different layers, allowing nodes to have different degree heterogeneity in different types of relations. Moreover, the latent positions $U = [U_1, \ldots, U_n]^\top \in \mathbb{R}^{n \times k}$ are shared between all layers, which capture the common structure between multiple networks among the same set of nodes. The node latent variables U enter the model through a layer-specific connection matrix $\Lambda^{(r)} \in \mathbb{R}^{k \times k}$, $r = 1, \ldots, R$. In general $\Lambda^{(r)} \in \mathbb{R}^{k \times k}$ does not need to be diagonal. In the special case when $\Lambda^{(r)} = I_k$, model (4.1) for a single layer coincides with the inner-product model considered in *Hoff* (2003) and *Ma and Ma* (2017). We propose to use non-diagonal $\Lambda^{(r)}$s as they allow not only different levels of homophily along different dimensions, but also general interactions between different dimensions of latent variables.

In summary, model (4.1) accommodates enough differences between layers, as it embeds each node through two components: layer-varying node individual effects $\{\alpha^{(r)}\}_{r=1}^R$ and layer-invariant latent positions U. Layer-specific connection matrices $\{\Lambda^{(r)}\}_{r=1}^R$ provide additional flexibility, allowing each layer to retain its own network-level characteristics. Therefore, information can be borrowed across different layers due to the shared latent structure, meanwhile each layer is also distinct in terms of its own node connecting behaviors.

Note in order for model (4.1) to be identifiable, additional constraints on parameters are necessary. The following proposition States the identifiability conditions, and its proof is provided in the Supplementary Material.

Proposition IV.1 (Identifiability conditions)**.** *Suppose that two sets of parameters* $(\{\alpha^{(r)}\}_{r=1}^R, \{\Lambda^{(r)}\}_{r=1}^R, U)$ *and* $(\{\alpha_\dagger^{(r)}\}_{r=1}^R, \{\Lambda_\dagger^{(r)}\}_{r=1}^R, U_\dagger)$ *satisfy the following conditions:*

A1. $J_n U = U$, $J_n U_\dagger = U_\dagger$, *where* $J_n = I_n - \frac{1}{n} 1_n 1_n^\top$;

A2. $U^\top U = nI_k$ *and* $U_\dagger^\top U_\dagger = nI_k$;

A3. At least one of $\Lambda^{(r)}$*'s,* $r = 1, 2, \cdots, R$, *is full rank.*

Then

$$\alpha^{(r)} 1_n^\top + 1_n \alpha^{(r)\top} + U\Lambda^{(r)} U^\top = \alpha_\dagger^{(r)} 1_n^\top + 1_n \alpha_\dagger^{(r)\top} + U_\dagger \Lambda_\dagger^{(r)} U_\dagger^\top$$

for $r = 1, \ldots, R$ *implies that there exists an orthonormal matrix* $O \in \mathbb{R}^{k \times k}$ *where* $O^\top O = OO^\top = I_k$, *such that*

$$\alpha_\dagger^{(r)} = \alpha^{(r)}, U_\dagger = UO, \Lambda_\dagger^{(r)} = O^\top \Lambda^{(r)} O,$$

for $r = 1, \cdots, R$.

4.3.2 Parameter Estimation

We define the objective function as the negative conditional log-likelihood of A under model (4.1):

$$\begin{aligned} & L\left(U, \{\alpha^{(r)}\}_{r=1}^R, \{\Lambda^{(r)}\}_{r=1}^R\right) \\ = & -\log P\left(A \middle| U, \{\alpha^{(r)}\}_{r=1}^R, \{\Lambda^{(r)}\}_{r=1}^R\right) \\ = & -\sum_{r=1}^R \sum_{i=1}^n \sum_{j=1}^n \left\{ A_{ij}^{(r)} \Theta_{ij}^{(r)} + \log\left(1 - \sigma(\Theta_{ij}^{(r)})\right) \right\}, \end{aligned} \tag{4.2}$$

where $\sigma(x) = 1/(1+\exp(-x))$ is the sigmoid fucntion. The goal is to find estimates $\widehat{U}$, $\{\widehat{\alpha}^{(r)}\}_{r=1}^R$, and $\{\widehat{\Lambda}^{(r)}\}_{r=1}^R$ that minimize the objective function defined in (4.2). For the purpose of interpretation and estimation, we treat all the parameters including the

node degree heterogeneity parameters $\{\alpha^{(r)}\}_{r=1}^R$ and latent positions U as fixed effects. This is different from the majority of existing work on single layer and multilayer network latent space models (*Hoff et al.*, 2002; *Hoff*, 2003; *Salter-Townshend and McCormick*, 2017; *D'Angelo et al.*, 2018; *DAngelo et al.*, 2019), where latent vectors U and node effects (if considered) are treated as random effects and Bayesian approaches are adopted for estimation. *Ma and Ma* (2017) is the first work that treated U as fixed latent representations and proposed a scalable projected gradient descent algorithm for estimating the single layer inner-product latent space model. In pursuit of computational efficiency, we adapt the projected gradient descent algorithm for estimating our multilayer network latent space model. Specifically, in each iteration, parameter estimates for U, $\{\alpha^{(r)}\}_{r=1}^R$ and $\{\Lambda^{(r)}\}_{r=1}^R$ are updated along the direction of their negative gradients of L and are further projected onto the set of parameter space that satisfies the identifiability condition. The procedure is summarized in Algorithm 4. Note that the update of U leverages the network links of all types. Therefore, we expect a superior estimation of U, in comparison to the estimate when using a single network only.

Algorithm 4 Projected Gradient Descent Algorithm for Parameter Estimation

Input: $A \in \mathbb{R}^{n\times n\times R}$; latent space dimension $k \geq 1$; initial estimates: $U_0, \{\alpha_0^{(r)}\}_{r=1}^R, \{\Lambda_0^{(r)}\}_{r=1}^R$; step sizes $\eta_u, \eta_\alpha, \eta_\lambda$; number of iterations T

Parameters:: $U, \{\alpha^{(r)}\}_{r=1}^R, \{\Lambda^{(r)}\}_{r=1}^R$

For $t = 0, 1, \ldots, T-1$

$U_{t+1} = U_t - \eta_u \nabla_U L = U_t + 2\eta_u \sum_{r=1}^R \left(A^{(r)} - \sigma(\Theta_t^{(r)})\right) U_t \Lambda^{(r)}$

$\alpha_{t+1}^{(r)} = \alpha_t^{(r)} - \eta_\alpha \nabla_{\alpha^{(r)}} L = \alpha_t^{(r)} + 2\eta_\alpha \left(A^{(r)} - \sigma(\Theta_t^{(r)})\right) 1_n,\ r = 1, \ldots, R$

$\Lambda_{t+1}^{(r)} = \Lambda_t^{(r)} - \eta_\lambda \nabla_{\Lambda^{(r)}} L = \Lambda_t^{(r)} + \eta_\lambda U_t^\top \left(A^{(r)} - \sigma(\Theta_t^{(r)})\right) U_t,\ r = 1, \ldots, R$

$U_{t+1} = J_n U_{t+1},\ U_{t+1} = U_{t+1} W$ for $W \in \mathbb{R}^{k\times k}$ s.t. $U_{t+1}^\top U_{t+1} = nI_k,\ \Lambda_{t+1}^{(r)} = (W^{-1})^\top \Lambda_{t+1}^{(r)} W^{-1}$

Output: $\widehat{U} = U_T, \widehat{\alpha}^{(r)} = \alpha_T^{(r)}, \widehat{\Lambda}^{(r)} = \Lambda_T^{(r)},\ r = 1, \ldots, R$

4.4 Theoretical Results

In this section we present our main theoretical results on the maximum likelihood estimators of model (4.1). Note though each $\Theta^{(r)}$ under model (4.1) is of rank at most $k+2$, the tensor that represents the overall edge connection probabilities, $[\Theta^{(1)};\ldots;\Theta^{(R)}]$, is beyond the low rank structure due to the layer-specific parameters $\{\alpha^{(r)}\}_{r=1}^{R}$. This brings non-trivial challenges for studying theoretical properties of estimators for the model, since most tensor recovery methods rely on a low-rank assumption for the tensor's mean structure (*Kolda and Bader*, 2009; *Wang and Song*, 2017; *Ghadermarzy et al.*, 2018; *Wang and Li*, 2018). We adopt recently developed tools in random tensor theory and tensor inequalities to establish an upper bound on the estimation error of $[\Theta^{(1)};\ldots;\Theta^{(R)}]$. Then using matrix perturbation theory, we further localize the overall error bound to a single network layer and apply the Davis-Kahan theorem to upper bound the estimation error of the common latent vectors U. Specifically, we first introduce the feasible parameter space as follows.

Definition IV.2. ***(Feasible parameter space).*** For $n, R, k \in \mathbb{N}, \mu \in \mathbb{R}_+$, the feasible parameter space $\mathcal{F} = \mathcal{F}_{n,R,k}(\mu)$ is defined as

$$
\begin{aligned}
\mathcal{F} &= \mathcal{F}_{n,R,k}(\mu) \\
&= \{\mathcal{T} = [\Theta^{(1)};\Theta^{(2)};\cdots;\Theta^{(R)}] \in \mathbb{R}^{n\times n\times R} : \\
&\Theta^{(r)} = \alpha^{(r)}1_n^\top + 1_n\alpha^{(r)\top} + U\Lambda^{(r)}U^\top; \\
&U \in \mathbb{R}^{n\times k}, U^\top U = nI_k, J_nU = U, \alpha^{(r)} \in \mathbb{R}^n, \\
&\Lambda^{(r)} \in \mathbb{S}^{k\times k}, \|\Theta^{(r)}\|_{\max} \le \mu, r = 1,2,\cdots,R\},
\end{aligned}
\tag{4.3}
$$

where $J_n = I_n - \frac{1}{n}1_n1_n^\top$, $\mathbb{S}^{k\times k}$ includes all symmetric $k\times k$ matrices, and $\|\cdot\|_{\max}$ represents the maximum absolute value of entries in a matrix.

Suppose the estimator $\widehat{\mathcal{T}}$ is obtained by

$$\widehat{\mathcal{T}} = \arg\min_{\mathcal{T}\in\mathcal{F}} L(\mathcal{T}), \tag{4.4}$$

where L is defined in (4.2). Theorem IV.3 provides the result on the error bound for $\widehat{\mathcal{T}}$.

Theorem IV.3. *Given the true parameters $\mathcal{T}_\star \in \mathcal{F}$, there exist absolute constants c_1, c_2, such that with probability at least $1 - R\exp(-c_1 n) - \exp(-c_2(2n+R))$, we have*

$$\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F^2 \leq C_1 nR + C_2(2n+R), \tag{4.5}$$

where C_1 and C_2 depend on μ and k.

The term $C_1 nR$ in (4.5) is induced by the layer specific parameters $\{\alpha^{(r)}\}_{r=1}^R$, and it grows linearly in the number of layers. The second term $C_2(2n+R)$ is induced by $\{U\Lambda^{(r)}U^\top\}_{r=1}^R$, and due to the common latent variables U among layers, the order of this term would not grow as fast as the first term as R grows. In the next theorem, we specify the relationship between the upper bound on the estimation error of U and the number of layers.

Theorem IV.4. *Denote $\{\alpha_\star^{(r)}\}_{r=1}^R$, $\{\Lambda_\star^{(r)}\}_{r=1}^R$, and $U_\star$ as the true parameters that form $\mathcal{T}_\star \in \mathcal{F}$. Assume that $\Lambda_\star^{(1)}, \Lambda_\star^{(2)}, \cdots, \Lambda_\star^{(R)}$ are of full rank, i.e.*

$$\sigma_{\min}(\Lambda_\star^{(r)}) \geq \kappa \; r = 1, 2, \cdots, R \tag{4.6}$$

for some constant $\kappa > 0$. Assume there exists a constant $\delta > 0$ such that $R \leq \delta n$, then with probability at least $1 - R\exp(-c_1 n) - \exp(-c_2(2n+R))$, we have

$$\min_{O: O^\top O = OO^\top = I_k} \left\{ \|\widehat{U} - U_\star O\|_F^2 \right\} \leq 8\kappa^{-2}(C_1 + \widetilde{C}_2 R^{-1}), \tag{4.7}$$

where $\widetilde{C}_2 = C_2(2+\delta)$ *and* c_1, c_2, C_1, C_2 *are the same constants as in Theorem IV.3.*

Theorem IV.4 demonstrates that the upper bound of the estimation error of U decreases inversely as the number of layers R grows. The constant term C_1 is induced from the layer-specific terms $\{\alpha^{(r)}\}_{r=1}^{R}$. In Section 4.5, we will numerically further demonstrate that under the regime $R = \mathcal{O}(n)$, the upper bound in (4.7) is inversely proportional to R.

Remark IV.5. Model (4.1) contains several tensor factorization models as special cases. For example, when $\alpha^{(r)} = 0$ for all $r = 1, \ldots, R$, it reduces to the logistic RESCAL model (*Nickel and Tresp*, 2013), for which theoretical properties were formerly unknown. The corollary below provides estimation property for latent factors under the logistic RESCAL model.

Corollary IV.6. *Assume* $\alpha^{(r)} = 0_n$, $r = 1, \ldots, R$ *for all* $\mathcal{T} \in \mathcal{F}$. *Under the same assumptions as in Theorem IV.4, as* $n \to \infty$, *with probability going to 1 we have*

$$\min_{O: O^\top O = OO^\top = I_k} \left\{ \|\widehat{U} - U_\star O\|_F^2 \right\} \leq \kappa^{-2} C R^{-1}$$

for an absolute constant C. *In other words, the upper bound of the estimation error of* U *decreases at the rate of* $\mathcal{O}(R^{-1})$.

Proofs of Theorem IV.3, Theorem IV.4 and Corollary IV.6 are all provided in the Appendix.

4.5 Simulation Studies

In this section, we investigate empirical performance of the proposed method by simulation studies. Specifically, we examine the estimation error of parameters with growing number of network layers. We also analyze the computational complexity of Algorithm 4.

We first study the relationship between the estimation error of the maximum likelihood estimators and the number of network layers. We set the true parameter values as follows.

- Generate $(U_\star)_{ij} \overset{iid}{\sim} \mathcal{N}(0,1)$ for $i = 1, \ldots, n$ and $j = 1, \ldots k$; transform $U_\star$ by 1) centering $U_\star$ s.t. $J_n U_\star = U_\star$, 2) rotating $U_\star$ s.t. $U_\star^\top U_\star \propto I_k$, and 3) scaling $U_\star$ s.t. $U_\star^\top U_\star = nI_k$.
- Generate $(\alpha_\star^{(r)})_i \overset{iid}{\sim} \text{Uniform}(-2,-1)$ for $i = 1, \ldots, n$ and $r = 1, \ldots, R$.
- Generate $\Lambda_\star^{(r)} = \text{diag}(\lambda_{\star,1}^{(r)}, \cdots, \lambda_{\star,k}^{(r)})$ where $\lambda_{\star,i}^{(r)} \overset{iid}{\sim} \text{Uniform}(-1,-0.5)$, for $r = 1, \ldots, R$.

Note that though $\Lambda_\star^{(r)}$'s are set to be diagonal, we do not require $\widehat{\Lambda}^{(r)}$'s to be diagonal when fitting the model.

We set $n = 400$, $R = 100$, and $k = 2$. More simulation results with $(n, R, k) = (200, 50, 2)$ and $(400, 100, 4)$ are provided in the Appendix. We generate 30 independent copies of the adjacency tensor A based on model (4.1). The first R_0 out of R layers are used to fit the model. We examine how the estimation errors of $\widehat{U}$ and $\{\widehat{\Theta}^{(r)}\}_{r=1}^{R_0}$ change with R_0. The estimation error of $\{\widehat{\Theta}^{(r)}\}_{r=1}^{R_0}$ is evaluated by the relative error

$$\left(\sum_{r=1}^{R_0} \|\widehat{\Theta}^{(r)} - \Theta_\star^{(r)}\|_F^2\right) \Big/ \left(\sum_{r=1}^{R_0} \|\Theta_\star^{(r)}\|_F^2\right), \tag{4.8}$$

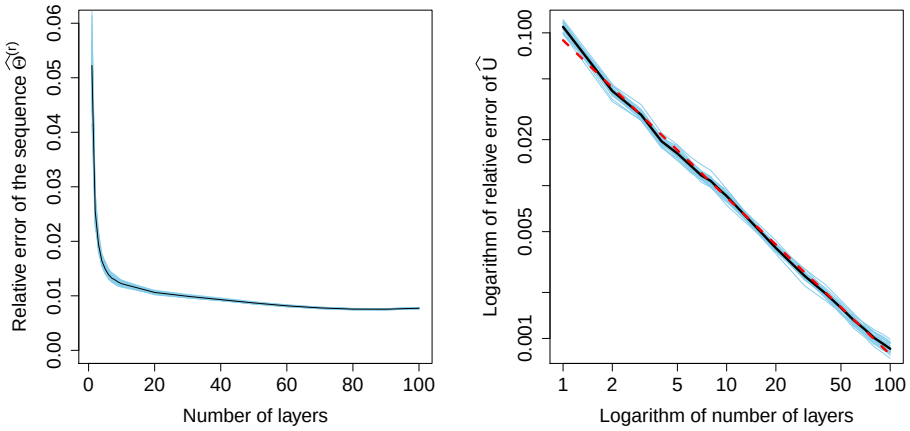
and the estimation error of $\widehat{U}$ is evaluated by

$$\min_{O: O^\top O = OO^\top = I_k} \left\{\|\widehat{U} - U_\star O\|_F^2\right\} / \|U_\star\|_F^2. \tag{4.9}$$

Finding the optimal O in (4.9) is known as the orthogonal Procrustes problem (*Schönemann*, 1966), which can be solved by singular value decomposition (SVD). In particular, denote the SVD of $\widehat{U}^\top U_\star$ be $S\Sigma V^\top$, then the optimal O is given by $VS^\top$.

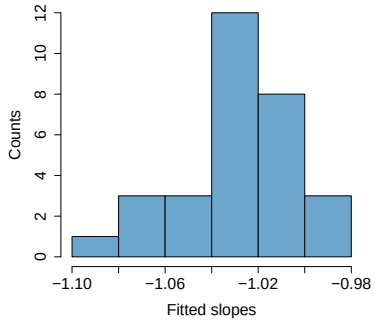
Note Algorithm 4 requires initialization of U_0, $\{\alpha_0^{(r)}\}_{r=1}^{R_0}$ and $\{\Lambda_0^{(r)}\}_{r=1}^{R_0}$. When

fitting the model, we initialize U_0 by first generating i.i.d. $\mathcal{N}(0,1)$ entries and then transforming it such that U_0 satisfies the identifiability condition. We initialize $\alpha_0^{(r)}$ as 0_n, i.e. the vector with all zeros, and we initialize $\Lambda_0^{(r)}$ as $\mathrm{diag}(-1,\ldots,-1)$. The step sizes $\eta_\alpha, \eta_\lambda$ are chosen to be small and fixed, and η_u is proportional to R_0^{-1}.



(a) Estimation error of $\{\widehat{\Theta}^{(r)}\}_{r=1}^{R_0}$

(b) Estimation error of $\widehat{U}$



(c) Histogram of fitted slopes

Figure 4.1: (a) and (b): Estimation error of parameters when $n = 400$, $R = 100$ and $k = 2$. Each light blue curve corresponds to one replication; the black curve corresponds to the average of all replications. The red dashed line in (b) corresponds to the line whose intercept and slope equal to the average of fitted intercepts and slopes respectively. (c): Histogram of all fitted slopes.

Figure 4.1(a) shows the estimation error of $\{\widehat{\Theta}^{(r)}\}_{r=1}^{R_0}$ given in (4.8) versus R_0. We can see as the number of layers grows, the relative error of $\{\widehat{\Theta}^{(r)}\}_{r=1}^{R_0}$ decreases and is

bounded below. By Theorem IV.3, we have

$$\sum_{r=1}^{R_0} \|\widehat{\Theta}^{(r)} - \Theta_\star^{(r)}\|_F^2 \leq C_1 n R_0 + C_2(2n + R_0) \tag{4.10}$$

with high probability. Also note that $\left(\sum_{r=1}^{R_0} \|\Theta_\star^{(r)}\|_F^2\right)$ is of order $\mathcal{O}(n^2 R_0)$ due to the constraints we put on the parameter space $\mathcal{F}$. Therefore, theoretically the bound of the relative error defined in (4.8) should be of order $\mathcal{O}\left(n^{-1} + n^{-1}R_0^{-1} + n^{-2}\right)$. For a fixed n, as R_0 increases, this term would decrease to some bound that depends on n^{-1}. The result in Figure 4.1(a) is then understandable as the "irreducible" estimation error of $\widehat{\Theta}^{(r)}$ comes from the first term in (4.10), i.e., the estimation error of layer-specific parameters $\widehat{\alpha}^{(r)}$, which does not decrease as the number of layers grows.

Figure 4.1(b) displays in log-log scale the estimation error of $\widehat{U}$ given by (4.9) against the number of network layers utilized for model fitting. For each replication, we fit a linear model to the result, i.e.,

$$\log(\text{relative error of } \widehat{U}) = a + b\log(R_0) + \epsilon.$$

Figure 4.1(c) shows the histogram of the fitted slopes. Note that all fitted slopes are close to -1, with the mean and standard deviation being -1.03 and 0.02 respectively. This agrees with the result in Theorem IV.4.

Since Algorithm 4 is a first-order method and aggregates gradient information of each layer in a linear manner, the running time should be proportional to R. Further, updating $\Theta_t^{(r)}$ in each iteration requires $\mathcal{O}(n^2 k)$ operations. Therefore the computational complexity of Algorithm 4 is $\mathcal{O}(n^2 R k)$. To verify this, we examine the computing time under two settings: 1) fixing n, increasing R, and 2) fixing R, increasing n. As shown in Figure 4.2, the running time per iteration is linear in R and quadratic in n.

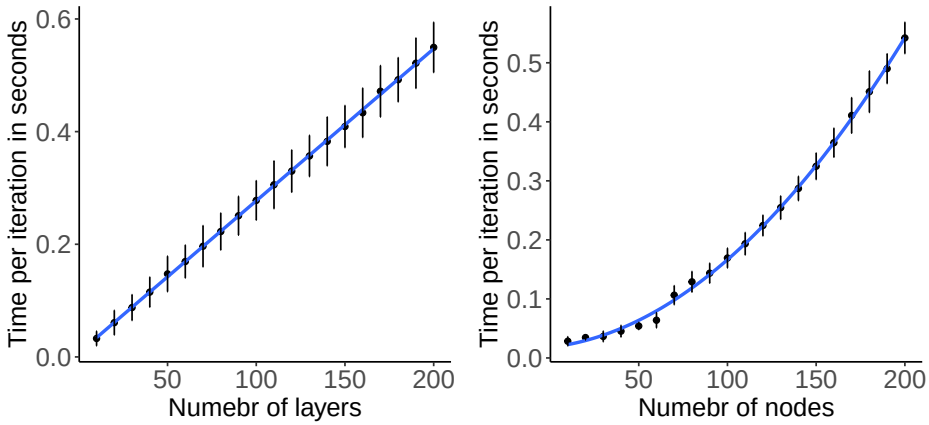


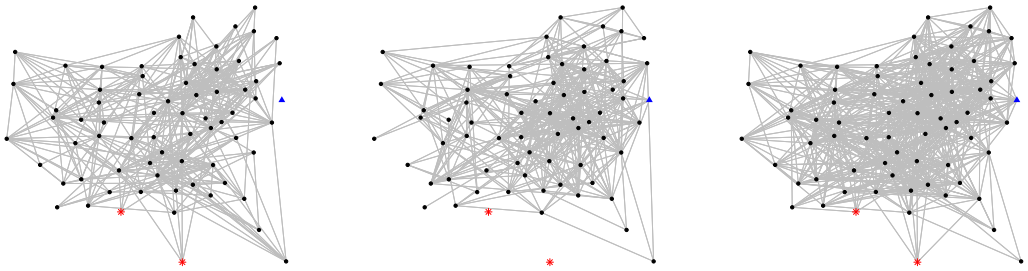
Figure 4.2: Average running time per iteration in seconds with one-standard-deviation error bars. Left: $n = 200$, and the R-square of a linear model is 0.998; Right: $R = 200$, and the R-square of a quadratic model is 0.998.

4.6 Real Data Applications

We apply the proposed model to two real-world examples. In practice, since there is no true value for the latent vectors, we can't evaluate the performance in terms of latent representation estimation. Instead, we consider two alternative approaches. First, though the latent vectors are not observed, there are usually observed node features which may be correlated with the latent vectors. Therefore, investigating the estimated latent representations against observed node features may provide insights on the estimation of latent vectors. Secondly, the estimated latent representations can often be used for downstream tasks, such as nodes classification, node clustering or link prediction. To examine the latent vector estimation, we demonstrate the performance of the proposed method on link prediction.

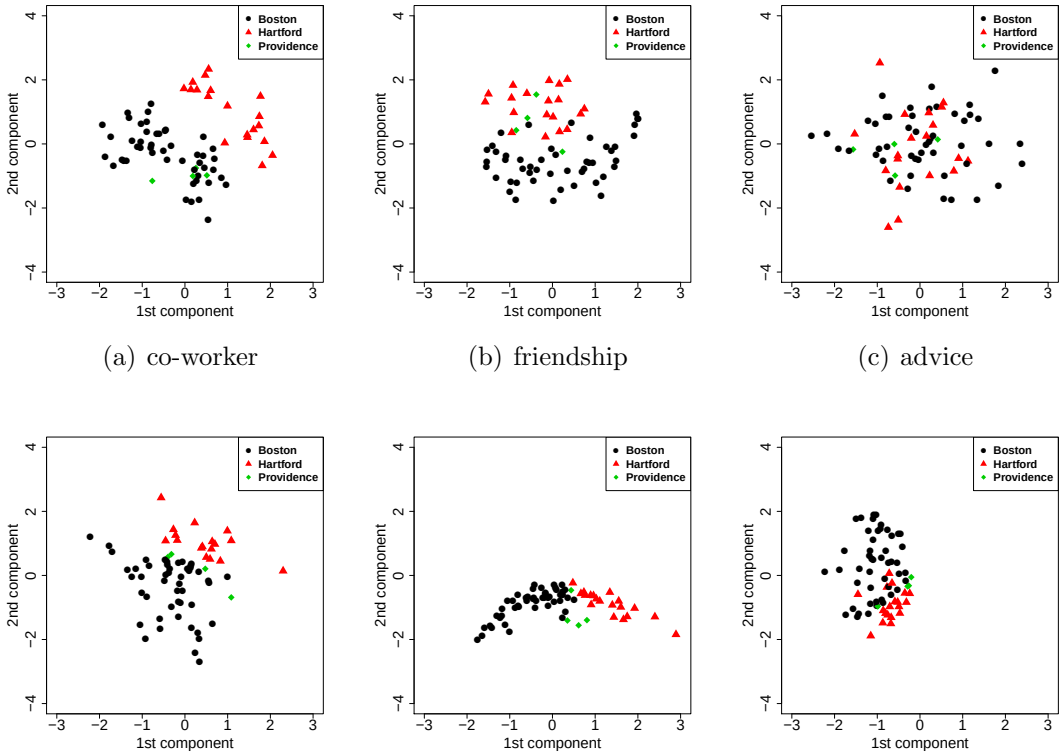
4.6.1 Lazega Lawyers Data

The Lazega Lawyers dataset records multiple connection relationships in a North-eastern US corporate law firm (*Lazega et al.*, 2001). There are three types of networks between 71 lawyers, which are their co-worker network, advice network, and friend-



(a) Co-worker network (b) Friendship network (c) Advice network

Figure 4.3: Visualization of the Lazega lawyer data. Nodes in different layers exhibit different connecting patterns. For example, the two red nodes are isolated in the friendship network but have several links in other layers, while the blue node is not connected to other nodes in the co-worker network but is well-connected in the friendship and advice networks.



(a) co-worker (b) friendship (c) advice

(d) proposed model (e) L-RESCAL model (f) COSIE

Figure 4.4: Upper row: estimated U based on single networks. Lower row: jointly estimated U based on multilayer networks using different methods. Color represents the lawyer's office.

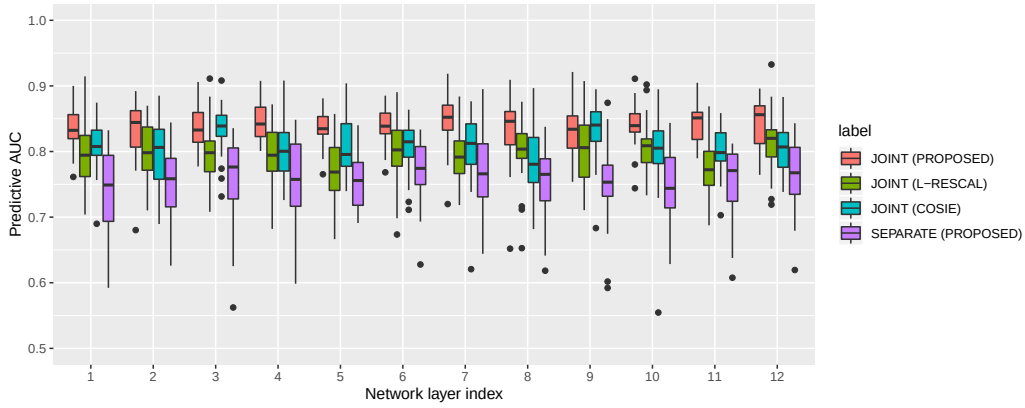


Figure 4.5: Link Prediction: AuROC on the test sets for 12 layers of the Karnataka Data.

ship network (Figure 4.3). The original network can be directed, for example, advice is often given in single direction and one nominates the other as a friend. We convert the direct networks to indirect ones by removing the directions. Besides the network relationships, multiple features of individuals are also recorded, including seniority, office, gender, law school attended, etc.

We fit both the multilayer version and single layer version of the proposed model (4.1). Initialization and stepsize choices are similar to what we do in Section 4.5. For comparison, we also fit model (4.1) without the layer-specific node individual effect terms, which reduces to the logistic RESCAL (L-RESCAL) model, as well as the COSIE model (*Arroyo et al.*, 2019), which utilizes the random dot-product model for multilayer networks and assumes that

$$\mathbb{E}[A^{(r)}] = U\Lambda^{(r)}U^\top, \ r = 1, \ldots, R.$$

We choose the dimension of the latent space to be $k = 2$ for the purpose of visualization. Figure 4.4 shows the estimated U from each model, and the colors are based on the lawyers' three offices: Boston, Hartford and Providence. From Figure 4.4, we can see more clear separation of people from different offices in the latent space based on the estimation from multilayer networks, in comparison to the estimation

using single networks. Moreover, comparing the proposed method to the other two multilayer network models which do not incorporate node individual effects, the proposed method shows the most clear separation in terms of the lawyers' offices, which, among the available node features, is presumably the most correlated one with all three types of networks.

4.6.2 Karnataka Data

In practice, estimation of the latent space model is often an initial step, and the estimated model can be further used for downstream tasks on networks, for example, link prediction. Suppose we are interested in predicting missing links in a target network. With multilayer networks, we investigate whether connections in other layers would assist in the prediction of links in the target layer, due to the correlation in network structures between different layers.

Banerjee et al. (2013) provided multiple social networks in villages in rural southern Karnataka, India. Within each village, 12 types of social relations are recorded, including borrow money from, give advice to, help with a decision, borrow kerosene or rice from, lend kerosene or rice to, lend money to, obtain medical advice from, engage socially with, are related to, go to temple with, invite to one's home, and visit in another's home. Some of the relations are directed, and as in Section 4.6.1, the directed networks are converted to indirected ones based on the existence of any single directional edge between nodes. The networks are collected at both individual level and household level. We analyzed the data on the household level and selected one representative village with 99 nodes. For each type of relation, we randomly remove 20% entries of the adjacency matrix as missing. For comparison, we fit the single layer version of the proposed model, the multilayer latent space model with and without layer-specific node individual effect terms, and the COSIE model using the observed entries. Then we predict link probabilities on those missing entries using

the fitted parameters and node latent representations. The dimension of the latent space is set to $k = 3$ for all methods. The experiments are replicated 30 times and we report AuROC for link prediction in Figure 4.5. As we can see, for each type of relation, using information from multiple networks has superior performances than using the layer itself only, which demonstrates that different layers share common structures and leveraging such information is beneficial. Moreover, the proposed method achieves the best performance in most layers, in comparison to the methods which do not take layer-specific node degree heterogeneity into account. This further supports the observed phenomenon that individual node behavior can vary from relation to relation, and modeling such flexibility is critical for capturing real-world network characteristics.

4.7 Conclusion and Discussion

In this paper, we have proposed a flexible and interpretable latent space model for multilayer networks. The proposed model is able to capture the common structure shared across different networks and meanwhile allows for heterogeneous layer-specific node connecting patterns. We have developed an efficient algorithm for parameter estimation. Moreover, theoretical guarantees on maximum likelihood estimators, in particular, improvements in the estimation of shared latent representations, are established. We have also demonstrated the proposed model on real-world data examples.

This work can be extended in several potential directions. Real-world networks are heterogeneous in the sense of not only multiple edge types, but also various node types (*Sun and Han*, 2013; *Huang et al.*, 2018; *Yu et al.*, 2018; *Zhang and Chen*, 2020; *Zitnik et al.*, 2018). One interesting direction is to extend the proposed modeling framework to networks with both multiple edge types and multiple node types, with each node type embedded into a unique latent space.

APPENDICES

APPENDIX A

Appendix of Chapter 2

A.1 Proof of Theorems

We introduce several notations that will be used throughout the proof. For any matrix $M \in \mathbb{R}^{p\times p}$, we denote $\varphi_{\max}(M)$ and $\varphi_{\min}(M)$ be its largest and smallest eigenvalues respectively. Let $\|M\|_F = \sqrt{\sum_{j,k} M_{jk}^2}$ be the Frobenious norm, and $\|M\|_2 = \varphi_{\max}(M)$ be the operator norm. Denote $|M|_{\max} = \max_{j,k} |M_{jk}|$ be the maximum of elements absolute values. For a vector v, we define $\|v\|_\infty = \max_j |v_j|$. Moreover, we denote $e_1, e_2, \ldots, e_p$ be the canonical basis for $\mathbb{R}^p$, and S^{p-1} represents a unit sphere.

A.1.1 Proof of Theorem II.7

Recall we view the data matrix $Z \in \mathbb{R}^{n\times q}$ as $Z = W + \epsilon$, where $W \sim \mathcal{MN}(0, I_q \otimes \Phi)$ and $\epsilon \sim \mathcal{MN}(0, \Sigma \otimes I_n)$ are independent. In the approximation algorithm, we use $\widehat{\Sigma} = Z^T Z/n - \widehat{tr}(\Phi) I_q/n$ to approximate $\epsilon^T\epsilon/n$. The main idea of the proof is first controlling the maximum value of $|\widehat{\Sigma} - \Sigma|$, then utilize the results in *Rothman et al.* (2008) to obtain the property of the estimator in (2.7).

Define $\Delta_1 = \widehat{\Sigma} - \Sigma$, then we have

$$\Delta_1 = \widehat{\Sigma} - \Sigma = \left(\frac{1}{n}\epsilon^T\epsilon - \Sigma\right) + \frac{1}{n}\left(W^T\epsilon + \epsilon^T W\right) + \frac{1}{n}\left(W^T W - tr(\Phi)I_q\right),$$

and

$$\begin{aligned} & |\Delta_1|_{\max} \\ \leq & \left|\frac{1}{n}\epsilon^T\epsilon - \Sigma\right|_{\max} + \frac{1}{n}\left|\epsilon^T W + W^T\epsilon\right|_{\max} + \frac{1}{n}\left|W^T W - tr(\Phi)I_p\right|_{\max} \qquad \text{(A.1)} \\ = & \text{I} + \text{II} + \text{III}. \end{aligned}$$

We bound the three terms in (A.1) respectively. Applying Lemma 1 in (*Ravikumar et al.*, 2011), we have

Lemma A.1. *Let $\epsilon \sim \mathcal{N}(0, \Sigma \otimes I_n)$, or ϵ_i be i.i.d $\mathcal{N}(0, \Sigma)$. Then, for (j, k), $1 \leq j \leq q$, $1 \leq k \leq q$,*

$$P\left[\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_{ij}\epsilon_{ik} - \Sigma_{jk}\right| \geq \delta\right] \leq 4\exp\left(-\frac{n\delta^2}{2c_0^2\max_j(\Sigma_{jj})^2)}\right),$$

for all $\delta \in (0, c_0\max_j(\Sigma_{jj}))$, wher c_0 is an absolute constant.

Lemma A.1 and the union sum inequality suggests the following corollary:

Corollary A.2. *There exists a constant c_1 such that with probability goes to 1,*

$$\text{I} = \left|\frac{1}{n}\epsilon^T\epsilon - \Sigma\right|_{\max} \leq c_1\sqrt{\frac{\log q}{n}},$$

where c_1 depends on c_0 and $\max_j(\Sigma_{jj})$ in Lemma A.1.

Corollary A.2 bounds term I in (A.1) . To bound II and III, we need the following lemma from *Rudelson and Zhou* (2017).

Lemma A.3. *(Lemma 32 in Rudelson and Zhou (2017)) Let $u, v \in S^{n-1}$. Let $M \succ 0$ be an $q \times q$ symmetric positive definite matrix. Let U be an $n \times q$ random matrix with*

independent entries U_{ij} satisfying $U_{ij} \sim \mathcal{N}(0,1)$. U_1, U_2 be independent copies of U. Then for every $t > 0$,

$$P\left(\left|u^T U_1 M^{1/2} U_2^T v\right| > t\right) \leq 2\exp\left(-c\min\left(\frac{t^2}{4tr(M)}, \frac{t}{2\|M\|_2^{1/2}}\right)\right),$$

and

$$P\left(\left|u^T U M^{1/2} U^T v - \mathbb{E}u^T U M^{1/2} U^T v\right| > t\right) \leq 2\exp\left(-c\min\left(\frac{t^2}{4\|M\|_F^2}, \frac{t}{2\|M\|_2}\right)\right),$$

where c is an absolute constant.

Then we have the following bounds on II, III and $|\Delta_1|_{\max}$:

Lemma A.4. *Assume the rank assumption for Φ (Assumption II.6) holds, then with probability goes to 1, there exist constants C_1, c_2, c_3 that depend on $\max_j \Sigma_{jj}$, $\overline{k}_1$ (defined in Assumption II.4) and c_Φ (defined in Assumption II.9), such that*

(a) $\text{II} = \frac{1}{n}|W^T\epsilon|_{\max} \leq c_2\sqrt{\frac{\log q}{n}}$;

(b) $\text{III} = \frac{1}{n}\left|W^T W - tr(\Phi)I_q\right|_{\max} \leq c_3\sqrt{\frac{\log q}{n}}$;

(c) $|\Delta_1|_{\max} \leq C_1\sqrt{\frac{\log q}{n}}$.

Proof. Note the fact that $\epsilon \sim U_1\Sigma^{1/2}$ and $W \sim \Phi^{1/2}U_2$, where U_1, U_2 are defined in Lemma A.3. Proof of parts (a) and (b) are based on proof of Lemma 11 in *Rudelson and Zhou* (2017).

(a) Let $e_1, e_2, \ldots, e_q \in \mathbb{R}^q$ be canonical basis spanning. Define $a_j = \frac{\Sigma^{1/2}e_j}{\|\Sigma^{1/2}e_j\|_2}$, $j = 1, \ldots, q$.

Define $t_1 = 2C_0\sqrt{\log q}tr(\Phi)^{1/2}$, where C_0 is properly chosen such that $cC_0^2 = 4$,

then applying Lemma A.3, we have

$$\begin{aligned}
&P\left(\exists j,k,\langle a_j, U_1^T\Phi U_2 e_k\rangle \geq t_1\right)\\
\leq&\sum_{j=1}^{q}\sum_{k=1}^{q} P(\langle a_j, U_1^T\Phi U_2 e_k\rangle \geq t_1))\\
\leq&2q^2\exp\left[-c\min\left(\frac{t_1^2}{K^4 tr(\Phi)}, \frac{t_1}{K^2\|\Phi\|_2^{1/2}}\right)\right]\\
\leq&2q^2\exp\left[-c\min\left(C_0^2, C_0\right)\log q\right]\\
\leq&2/q^2.
\end{aligned} \tag{A.2}$$

So with probability at least $1-2/q^2$,

$$\begin{aligned}
&|W^T\epsilon|_{\max} = |\epsilon^T W|_{\max} = |\Sigma^{1/2}U_1^T\Phi^{1/2}U_2|_{\max}\\
=&\max_j \|\Sigma^{1/2}U_1^T\Phi^{1/2}U_2 e_j\|_\infty = \max_j\max_k |e_k\Sigma^{1/2}U_1^T\Phi^{1/2}U_2 e_j|\\
=&\max_{j,k}\langle e_k\Sigma^{1/2}, U_1^T\Phi^{1/2}U_2 e_j\rangle = \max_{j,k}\|e_k\Sigma^{1/2}\|_2\langle a_k, U_1^T\Phi^{1/2}U_2 e_j\rangle\\
\leq&\max_j \Sigma_{jj}^{1/2} C_0 K^2\sqrt{\log q}\ \mathrm{tr}(\Phi)^{1/2}.
\end{aligned} \tag{A.3}$$

The last inequality is due to Assumption II.6.

(b) Define $t_2 = C_0K^2\sqrt{\log q}\|\Phi\|_F$, then

$$\begin{aligned}
&P\left(\exists j,k,\langle e_k, (W^TW - tr(\Phi)I_q)e_j\rangle \geq t_2\right)\\
\leq&2q^2\exp\left[-c\min\left(\frac{t_2^2}{K^4\|\Phi\|_F^2}, \frac{t_2}{K^2\|\Phi\|_2}\right)\right]\\
\leq&2/q^2.
\end{aligned} \tag{A.4}$$

So with probability $1-2/q^2$,

$$
\begin{aligned}
&|W^TW - tr(\Phi)I_p|_{max} \\
=&\max_{j,k}\langle e_k, (W^TW - tr(\Phi)I_p)e_j\rangle = \max_{j,k}\langle e_k, (U^T\Phi U - tr(\Phi)I_p)e_j\rangle \\
\leq& 2C_0\sqrt{\log p}\|\Phi\|_F \leq 2C_0\sqrt{\log p}tr(\Phi)^{1/2}\|\Phi\|_2^{1/2} \\
\leq& 2C_0\sqrt{\log p}tr(\Phi)^{1/2}\overline{k}_1^{1/2}.
\end{aligned} \tag{A.5}
$$

The last inequality is due to Assumption II.6.

(c) (a) and (b) shows with probability $1 - 4/q^2 \to 1$,

$$
\text{II} + \text{III} \leq (c_2 + c_3)\sqrt{\frac{\log q}{n}},
$$

for $c_2 = 2\max_j \Sigma_{jj}^{1/2} C_0 K^2 c_\Phi$ and $c_3 = C_0 K^2 \overline{k}_1^{1/2} c_\Phi$.

Combined with results from Corollary A.2, we have with probability goes to 1,

$$
|\Delta_1|_{\max} < C_1\sqrt{\frac{\log q}{n}},
$$

where $C_1 = c_1 + c_2 + c_3$. □

Corollary A.2 and Lemma A.4 leads to the main result in Theorem II.7. The proof is given below.

Proof of Theorem 1. The proof of of Theorem 1 in (*Rothman et al.*, 2008) implies that as long as $\widehat{\Sigma}$ in (2.5) satisfy the condition that with probability tending 1, there exists C_1 such that $|\widehat{\Sigma} - \Sigma|_{max} \leq C_1\sqrt{\log q/n}$, then the solution from (2.5) would satisfy the property that

$$
\|\widehat{\Omega}_\lambda - \Omega_0\|_F = O_P\left(\sqrt{\frac{(q+s)\log q}{n}}\right),
$$

where s bounds the number of non-zero entries in Ω_0. The proof procedure is similar

and we omit it here. □

A.1.2 Proof of Theorem II.11

As in the proof in Section A.1.1, we first establish the bound for the maximum of $|\widehat{\Phi} - \Phi|$, then we give the consistency of $\widehat{\Theta}$.

Define $\Delta_2 = \widehat{\Phi} - \Phi$, we have

$$\begin{aligned}\widehat{\Phi} &= \frac{ZZ^T}{q} - \frac{\widehat{tr}(\Sigma)}{q} I_n \\ &= \left(\frac{1}{q} WW^T - \Phi\right) + \frac{1}{q}(W\epsilon^T + \epsilon W^T) + \frac{1}{q}(\epsilon\epsilon^T - tr(\Sigma) I_n) + \frac{1}{q}(tr(\Sigma) I_n - \widehat{tr}(\Sigma) I_n).\end{aligned}$$

Then

$$\begin{aligned}|\Delta_2|_{\max} &\leq \left|\frac{1}{q} WW^T - \Phi\right|_{\max} + \frac{1}{q}\left|W\epsilon^T + \epsilon W^T\right|_{\max} \\ &\quad + \frac{1}{q}\left|\epsilon\epsilon^T - tr(\Sigma) I_n\right|_{\max} + \frac{1}{q}\left|tr(\Sigma) I_n - \widehat{tr}(\Sigma) I_n\right|_{\max} \\ &= \mathrm{I} + \mathrm{II} + \mathrm{III} + \mathrm{IV}.\end{aligned}$$

For simplicity of notations, we keep using $\mathrm{I}, \mathrm{II} \ldots$ to represent each part that we need to bound, but these are different from those used in Section A.1.1.

Since $W \sim \mathcal{MN}(0, \Phi_{n\times n} \otimes I_q)$, so similarly to Lemma A.1 and Corollary A.2, we have

Lemma A.5. *For (i, i'), $1 \leq i \leq n, 1 \leq i' \leq n$,*

$$P\left[\left|\frac{1}{q}\sum_{j=1}^{q} W_{ij} W_{i'j} - \Phi_{ii'}\right| \geq \delta\right] \leq 4\exp\left(-\frac{q\delta^2}{2(c_0')^2 \max_i(\Phi_{ii})^2)}\right)$$

for all $\delta \in (0, c_0' \max_i(\Phi_{ii}))$, where c_0' is an absolute constant.

Taking the union bound over all (i, i'), we have

Corollary A.6. *With probability goes to 1, there exists a constant c_1' such that*

$$\left|\frac{1}{q}\sum_{j=1}^{q} W_{ij}W_{i'j} - \Phi_{ii'}\right|_{\max} \leq c_1'\sqrt{\frac{\log n}{q}},$$

where c_1' depends on c_0' and $\max_i(\Phi_{ii})$ in Lemma A.5.

Bounding II and III is also similar to part(a) and part(b) in Lemma A.4. We only need to exchange U_1 and U_2, W and ϵ, and n and q in the proof, and we have the following results:

Lemma A.7. *Assume Assumption II.10 holds, then with probability goes to 1, there exist constant c_2', c_3' that depend on $\max_i \Phi_{ii}$, $\overline{k}_2$ (defined in Assumption II.8) and $\overline{c}_1$ (defined in Assumption II.5), such that*

(a) $\text{II} = \frac{1}{q}\left|W\epsilon^T + \epsilon W^T\right|_{\max} \leq c_2'\sqrt{\frac{\log n}{q}}$;

(b) $\text{III} = \frac{1}{q}\left|\epsilon\epsilon^T - tr(\Sigma)I_n\right|_{\max} =\leq c_3'\sqrt{\frac{\log n}{q}}$.

A bound for IV has been given in Lemma 5 in *Rudelson and Zhou* (2017).

Lemma A.8. *Let $n > 2$. Let $\widehat{tr}(\Sigma) = \frac{1}{n}\left(\|Z\|_F^2 - qtr(\Phi)\right)_+$. With probability greater than $1 - 1/q^3$, we have*

$$\frac{1}{q}\left|tr(\Sigma) - \widehat{tr}(\Sigma)\right| \leq 4C_0\sqrt{\frac{\log q}{qn}}\left(\frac{\|\Sigma\|_F}{\sqrt{q}} + \frac{\|\Phi\|_F}{\sqrt{n}}\right)$$

where C_0 is absolute constants.

By Assumption II.6 and II.10, $\|\Phi\|_F/\sqrt{n} \leq \|\Phi\|_2$ and $\|\Sigma\|_F/\sqrt{q} \leq \|\Sigma\|_2$. So we further have:

Lemma A.9. *With probability goes to 1,*

$$\frac{1}{q}\left|tr(\Sigma) - \widehat{tr}(\Sigma)\right| \leq c_4'\sqrt{\frac{\log q}{n}},$$

where $c_4' = \frac{2}{q} C_0 K^2 (\|\Phi\|_2 + \|\Sigma\|_2)$.

Based on Lemma A.7 and Lemma A.9, we obtain the bound on $|\Delta_2|_{\max}$:

Lemma A.10. *With probability goes to 1, we have* $|\Delta_2|_{\max} \leq C_1' \sqrt{\frac{\log n}{q}}$, *where* $C_1' = c_1' + c_2' + c_3' + c_4'$, *where the constants are given in Lemma A.7 and Lemma A.9.*

Note that if

$$\widehat{\Theta} = \arg \min_{\substack{\Theta \succeq 0 \\ \Theta_{ij}=0, (i.j) \notin \mathcal{E}}} \left\{ tr\Theta\widehat{\Phi} - \log|\Theta| \right\},$$

then the solution will satisfy

$$\begin{aligned}
\widehat{\Theta}_{ij} &= 0, \forall (i.j) \notin \mathcal{E}, \\
\widehat{\Theta}_{ij}^{-1} &= \widehat{\Phi}_{ij}, \forall (i.j) \in \mathcal{E}, \\
\widehat{\Theta}_{ii}^{-1} &= \widehat{\Phi}_{ii}, \forall i = 1, \ldots, n.
\end{aligned}$$

Therefore, to ensure the solution of (2.8) satisfy the identifiability condition, i.e., the diagonal elements equal to a known constant, we need to first rescale $\widehat{\Phi}$ before plugging it into (2.8). In the next lemma, we shows that the difference between rescaled $\widehat{\Phi}$ and Φ is also bounded. Without loss of generality, we consider the case when $c_\Phi = 1$.

Lemma A.11. *Denote the rescaled matrix by* $\widehat{\Phi}_0$, *where* $\widehat{\Phi}_{0,ij} = \widehat{\Phi}_{ij} / \sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}}$. *Then with probability goes to 1, there exists constant* C_1'' *s.t.* $|\widehat{\Phi}_0 - \Phi|_{\max} \leq C_1'' \sqrt{\log n / q}$.

Proof. For any (i, j),

$$\begin{aligned}
\left| \frac{\widehat{\Phi}_{ij}}{\sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}}} - \Phi_{ij} \right| &= \left| \frac{1}{\sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}}} \left(\widehat{\Phi}_{ij} - \Phi_{ij} \right) + \left(1 - \frac{1}{\sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}}} \right) \Phi_{ij} \right| \\
&\leq \left| \frac{1}{\sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}}} \left(\widehat{\Phi}_{ij} - \Phi_{ij} \right) \right| + \left| \left(1 - \frac{1}{\sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}}} \right) \Phi_{ij} \right|.
\end{aligned}$$

Lemma A.10 implies that $|\widehat{\Phi}_{ii} - \Phi_{ii}| = O_P\left(\sqrt{\log n/q}\right)$, so $\widehat{\Phi}_{ii} = O_P(1)$. Thus $\left(\sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}}\right)^{-1} = O_P(1)$ and $\left(\widehat{\Phi}_{ij} - \Phi_{ij}\right)/\sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}} = O_P\left(\sqrt{\log n/q}\right)$.

Further, Lemma A.10 suggests that $1 - C_1'\sqrt{\log n/q} < \widehat{\Phi}_{ii} < 1 + C_1'\sqrt{\log n/q}$, so does for $\widehat{\Phi}_{jj}$. So

$$\frac{1}{1 + C_1'\sqrt{\dfrac{\log n}{q}}} \leq \frac{1}{\sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}}} \leq \frac{1}{1 - C_1'\sqrt{\dfrac{\log n}{q}}},$$

and

$$\frac{1}{1 \pm C_1'\sqrt{\dfrac{\log n}{q}}} = 1 \mp C_1'\sqrt{\frac{\log n}{q}} + o\left(\sqrt{\frac{\log n}{q}}\right).$$

Thus, $\left|1 - 1/\sqrt{\widehat{\Phi}_{ii}\widehat{\Phi}_{jj}}\right| = O_P\left(\sqrt{\log n/q}\right)$. The bounds on both parts suggests there exists constant C_1'' such that $|\widehat{\Phi}_0 - \Phi|_{\max} \leq C_1''\sqrt{\log n/q}$. □

In implementation, we will plug in the rescaled $\widehat{\Phi}$ into (2.8). The above results lead to the consistency of estimator obtained from (2.8) as stated in Theorem II.11. The proof is given below.

Proof of Theorem 2. The proofs are based on *Rothman et al.* (2008) and *Zhou et al.* (2011).

Denote $\Theta_{\mathcal{E}} = \{\Theta \in \mathbb{R}^{n\times n} | \Theta \succeq 0, \Theta_{ij} = 0, (i.j) \notin \mathcal{E}\}$. Let $\widehat{\Theta}$ be the solution to

$$\widehat{\Theta} = \arg\min_{\Theta\in\Theta_{\mathcal{E}}} \left\{ tr\left(\Theta\widehat{\Phi}\right) - \log|\Theta| \right\}.$$

We have shown that with probability goes to 1, there exists constant C_1'' such that $\left|\widehat{\Phi} - \Phi\right|_{\max} \leq C_1''\sqrt{\log n/q}$. Denote $\Theta_0 = \Phi^{-1}$ as the truth precision matrix, and

define

$$
\begin{aligned}
Q(\Theta) &= tr\left(\Theta\widehat{\Phi}\right) - \log|\Theta| - tr\left(\Theta_0\widehat{\Phi}\right) - \log|\Theta_0| \\
&= tr\left[(\Theta-\Theta_0)(\widehat{\Phi}-\Phi)\right] - (\log|\Theta| - \log|\Theta_0|) + tr\left[(\Theta-\Theta_0)\Phi\right]
\end{aligned}
$$

for $\Theta \in \Theta_{\mathcal{E}}$. Since $\widehat{\Theta}$ minimizes $Q(\Theta)$, then $\widehat{\Delta} = \widehat{\Theta} - \Theta_0$ minimizes $G(\Delta) \coloneqq Q(\Theta_0 + \Delta)$. Moreover, $G(\Delta) \leq G(0) = 0$. We define $\Theta_M = \left\{\Delta : \Delta = \Delta^T, \|\Delta\|_F = Mr_p\right\}$, where $r_q = \sqrt{(n+2|E|)\log n/q}$. It suffices to show that $\inf_{\Delta\in\Theta_M} G(\Delta) > 0$, therefore $\widehat{\Delta}$ must be inside the sphere defined by Θ_M and $\|\Delta\|_F \leq M\sqrt{(n+2|E|)\log n/q}$.

It has been shown in *Rothman et al.* (2008) that

$$
\begin{aligned}
&\log|\Phi^{-1}+\Delta| - \log|\Phi^{-1}| \\
=&tr(\Phi\Delta) - vec(\Delta)^T\left(\int_0^1 (1-\nu)(\Phi^{-1}+\nu\Delta)^{-1}\otimes(\Phi^{-1}+\nu\Delta)^{-1}d\nu\right)vec(\Delta) \\
=&tr(\Phi\Delta) - C_\Delta.
\end{aligned}
$$

Also it was shown that $C_\Delta \geq \frac{1}{4}k_2^2\|\Delta\|_F^2$. Moreover,

$$
|tr\Delta(\widehat{\Phi}-\Phi)| \leq |\sum_{i\neq i'}(\widehat{\Phi}_{ii'}-\Phi_{ii'})\Delta_{ii'}| + |\sum_i(\widehat{\Phi}_{ii}-\Phi_{ii})\Delta_{ii}|. \tag{A.6}
$$

The first summand in (A.6) could be bounded by $C_1''\sqrt{\log n/q}\|\Delta\|_{1,off}$. Note that since $\Theta \in \Theta_{\mathcal{E}}$, then $\Delta_{i,i'} = 0$ for $(i,i') \notin \mathcal{E}$. So $\|\Delta\|_{1,off} = \|\Delta\|_{\mathcal{E}} = 2\sum_{(i,i')\in\mathcal{E}}\Delta_{ii'} \leq \sqrt{2|E|}\|\Delta_{\mathcal{E}}\|_F \leq \sqrt{2|E|}\|\Delta\|_F$. For the second summand in (A.6), since $\Delta_{ii} = 0$ by the constraints we put on diagonal elements of Φ and $\widehat{\Theta}^{-1}$, so the term vanishes.

Therefore, we have

$$\begin{aligned} G(\Delta) &= tr(\Delta(\widehat{\Phi} - \Phi)) + C_\Delta \\ &\geq \frac{1}{4}\underline{k}_2^2\|\Delta\|_F^2 - C_1''\sqrt{\frac{\log n}{q}}|\Delta|_{1,off} \\ &\geq \frac{1}{4}\underline{k}_2^2\|\Delta\|_F^2 - C_1''\sqrt{\frac{2|E|\log n}{q}}\|\Delta\|_F \\ &= \|\Delta\|_F^2\left(\frac{1}{4}\underline{k}_2^2 - C_1''\sqrt{\frac{2|E|\log n}{q}}\|\Delta\|_F^{-1}\right) \\ &= \|\Delta\|_F^2\left(\frac{1}{4}\underline{k}_2^2 - \frac{C_1''}{M}\right). \end{aligned}$$

$G(\Delta) > 0$ for M sufficiently large, and the proof is completed. □

A.2 EM algorithm for Parameter Estimation

A.2.1 Estimation for Zero-mean Matrix Variate Model

Consider the matrix variate model with zero mean and Kronecker sum covariance:

$$Z_{n\times q} \sim \mathcal{MN}(0_{n\times q}, \Sigma_{q\times q} \oplus \Phi_{n\times n}).$$

Estimating Σ and Φ directly by maximizing the marginal likelihood of $Z_{n\times q}$ is difficult, since the log-likelihood involves analytical solution of the inverse of Kronecker sum of two matrices. Based on our interpretation of the model that Z could be viewed as the sum of two independent components, $Z_{n\times q} = W_{n\times q} + \epsilon_{n\times q}$, we consider using standard EM algorithm to estimate Σ and Φ by treating $W_{n\times q}$ as latent variables.

E-step

We first calculate the log-likelihood of complete data (Z, W). Since $Z_{n\times q}|W_{n\times q} \sim \mathcal{MN}(W_{n\times q},$

$\Sigma_{q\times q} \otimes I_n$), and $W_{n\times q} \sim \mathcal{MN}(0, I_p \otimes \Phi_{n\times n})$, then the log likelihood is (up to scaling and additive constants):

$$\begin{aligned}
&l_c(\Sigma, \Phi|Z, W) = \log P(W|\Phi) + \log P(Z|W, \Sigma) \\
&= -q \log|\Phi| - vec(W)^T (I_p \otimes \Phi)^{-1} vec(W) \\
&- n \log|\Sigma| - vec(Z - W)^T (\Sigma \otimes I_n)^{-1} vec(Z - W) \\
&= -q \log|\Phi| - tr(W^T \Phi^{-1} W\big) - n \log|\Sigma| - tr\big((Z - W)\Sigma^{-1}(Z - W)^T\big) \\
&= -q \log|\Phi| - tr(\Phi^{-1} W W^T\big) - n \log|\Sigma| - tr\big(\Sigma^{-1}(Z - W)^T (Z - W)\big).
\end{aligned}$$

Next we calculate the distribution of latent variable $W_{n\times q}$ conditional on observed data $Z_{n\times q}$:

$$\begin{aligned}
&P(Z|W) \propto P(W)P(Z|W) \\
&\propto \exp\Big\{ -\frac{1}{2} vec(W)^T (I_q \otimes \Phi)^{-1} vec(W) \\
&-\frac{1}{2}\Big(vec(Z) - vec(W)\Big)^T (\Sigma \otimes I_n)^{-1}\Big(vec(Z) - vec(W)\Big)\Big\} \\
&\propto \exp\Big\{ -\frac{1}{2} vec(W)^T (I_q \otimes \Phi^{-1} + \Sigma^{-1} \otimes I_n) vec(W) + \Big(vec(Z)^T (\Sigma^{-1} \otimes I_n)\Big) vec(W)\Big\}.
\end{aligned}$$

Denote $\Omega_W = I_q \otimes \Phi^{-1} + \Sigma^{-1} \otimes I_n$, then the last expression is proportional to

$$\exp\Big\{ -\frac{1}{2}\Big(vec(W) - \Omega_W^{-1}(\Sigma^{-1} \otimes I_n) vec(Z)\Big)^T \Omega_W \Big(vec(W) - \Omega_W^{-1}(\Sigma^{-1} \otimes I_n) vec(Z)\Big)\Big\}.$$

Therefore, the conditional distribution $W_{n\times q}|Z_{n\times q}$ is

$$W|Z \sim \mathcal{MN}(\mu_W, \Sigma_W),$$

where

$$\Sigma_W = \Omega_W^{-1} = \big(I_q \otimes \Phi^{-1} + \Sigma^{-1} \otimes I_n\big)^{-1},$$

and

$$vec(\mu_W) = \Sigma_W\Big((\Sigma^{-1} \otimes I_n)vec(Z)\Big).$$

Next we take expectation of $l_c(\Sigma, \Phi|Z, W)$ with respect to $W|Z$ and obtain:

$$\begin{aligned} E_{W|Z} l_c(\Sigma, \Phi|Z, W) = &-q \log|\Phi| - tr\left(\Phi^{-1} E_{W|Z} W W^T\right) \\ &- n \log|\Sigma| - tr\left(\Sigma^{-1} E_{W|Z}(Z-W)^T(Z-W)\right). \end{aligned}$$

We first calculate $E_{W|Z} W W^T$:

$$\begin{aligned} \mathbb{E}_{W|Z} W W^T &= \mathbb{E}_{W|Z}(\sum_{j=1}^{q} W_{\cdot j}(W_{\cdot j})^T) \\ &= \sum_{j=1}^{q}\Big((\mu_W)_{\cdot j}((\mu_W)_{\cdot j})^T + cov_{W|Z}\big(W_{\cdot j}, (W_{\cdot j})^T\big)\Big) = \mu_W \mu_W^T + \sum_{j=1}^{q}(\Sigma_W)_{jj}. \end{aligned}$$

Here Σ_W is of size $nq \times nq$, it could be written as $q \times q$ blocks with each block of size $n \times n$:

$$\Sigma_W = \begin{bmatrix} (\Sigma_W)_{11} & \dots & (\Sigma_W)_{1q} \\ \vdots & \ddots & \vdots \\ (\Sigma_W)_{q1} & \dots & (\Sigma_W)_{qq} \end{bmatrix}.$$

Next we calculate $\mathbb{E}_{W|Z}(Z-W)^T(Z-W)$. For (i, j), $1 \le i, j \le q$:

$$(\mathbb{E}_{W|Z} W^T W)_{ij} = \mathbb{E}_{W|Z}((W_{\cdot i})^T W_{\cdot j}) = ((\mu_W)_{\cdot i})^T (\mu_W)_{\cdot j} + tr\Big((\Sigma_W)_{ij}\Big),$$

where $(\Sigma_W)_{ij}$ is the (i, j)th block of Σ_W. So

$$\mathbb{E}_{W|Z} W^T W = \mu_W^T \mu_W + tr_n(\Sigma_W),$$

where $tr_n(\Sigma_W)$ is a $q \times q$ matrix that takes trace of each $n \times n$ block of Σ_W:

$$tr_n(\Sigma_W) = \begin{bmatrix} tr((\Sigma_W)_{11}) & \dots & tr((\Sigma_W)_{1q}) \\ \vdots & \ddots & \vdots \\ tr((\Sigma_W)_{q1}) & \dots & tr((\Sigma_W)_{qq}) \end{bmatrix}.$$

In summary, in E-step, we have

$$\mathbb{E}_{W|Z} l_c(\Phi, \Sigma | Z, W) = -q \log |\Phi| - tr\Big(\Phi^{-1}\big((\mu_W \mu_W^T + \sum_{j=1}^{q} (\Sigma_W)_{jj}\big)\Big) \\ - n \log |\Sigma| - tr\Big(\Sigma^{-1}\big((Z - \mu_W)^T (Z - \mu_W) + tr_n(\Sigma_W)\big)\Big).$$

M-step

In M-step, we maximize $\mathbb{E}_{W|Z} l_c(\Phi, \Sigma | Z, W)$ with respect to Σ and Φ. Note that Σ and Φ are not coupled together in $\mathbb{E}_{W|Z} l_c(\Phi, \Sigma | Z, W)$, so we can maximize them separately. For notation simplicity, we denote $S_1 = \Sigma^{-1}\big((Z - \mu_W)^T (Z - \mu_W) + tr_n(\Sigma_W)\big)/n$ and $S_2 = \big((\mu_W \mu_W^T + \sum_{j=1}^{q} (\Sigma_W)_{jj}\big)/q$. We consider the situation for both $n \gg q$ and $n \not\gg q$.

Low dimensional setting When $n \gg q$, we could not afford too many parameters in $\Phi_{n \times n}$. So we require Φ takes the specific form: $\Phi^{-1} = \tau^{-2}(L + \gamma I_n)$. We only estimate τ^2 and view γ as a known or a tuning parameter. In M-step, we update τ^2 and Σ respectively by:

$$\Sigma \leftarrow S_1,$$

$$\tau^2 \leftarrow tr\left((L + \gamma I)^{-1} S_2\right)/n.$$

High dimensional setting When $n \not\gg q$, we assume $\Phi_{ii} = c_\Phi$ for $i = 1, 2, \dots, n$ with a known constant c_Φ and $tr(\Phi) = c_\Phi n$ for identifiability issue. Further, we assume Σ^{-1} satisfy a pre-specified sparsity level, for example, we assume $\|\Sigma^{-1}\|_{1,off} \leq$

C_1 for some constant C_1. Maximizing the quantity in $\mathbb{E}_{W|Z} l_c(\Phi, \Sigma|Z, W)$ with sparsity constraint could be solved by existing tools, e.g., graphical lasso. Under the high-dimensional setting, we update Σ and Φ by:

$$\widehat{\Sigma} = \arg\min_{\Sigma \succeq 0} \left\{ \mathrm{tr}\left(\Sigma^{-1} S_1\right) + \log|\Sigma| + \rho \|\Sigma^{-1}\|_{1,\mathrm{off}} \right\}, \tag{A.7}$$

$$\widehat{\Phi} = \arg \min_{\substack{\Phi \succeq 0 \\ \Phi_{ij}^{-1}=0,(i,j)\notin\mathcal{E}}} \left\{ \mathrm{tr}\left(\Phi^{-1} S_2\right) + \log|\Phi| \right\}. \tag{A.8}$$

Here ρ can be chosen over a grid such that the updated $\widehat{\Sigma}$ satisfy the pre-specified sparsity. In practice, we may also add a small penalty in (A.8) for numerical stability. At each step we rescale $\widehat{\Phi}$ such that $tr(\widehat{\Phi})$ equals to the pre-specified number, ensuring the estimated Φ satisfy the identifiability condition.

A.2.2 Extension to Classification Setting

For the classification setting, we have

$$X_{n\times p}|Y_{n\times 1} \sim \mathcal{MN}(M_{n\times p}, \Sigma_{p\times p} \oplus \Phi_{n\times n})$$

where $M_{i\cdot} = \sum_{k=1}^{K} \mathbf{1}(Y_i = k)\mu_k$, with μ_k is a row vector in $\mathbb{R}^p$ representing the mean parameter for class k. $\mu_{K\times p} = [\mu_1^T, \mu_2^T, \ldots, \mu_K^T]^T$. We need to estimate Σ, Φ and μ. In this subsection we modify the EM algorithm in Section A.2.1. When X has a non-zero mean, we could still view X as summation of two independent parts, $X = W + \epsilon$, where $W \sim \mathcal{MN}(M_{n\times p}, I_p \times \Phi)$ and $\epsilon \sim \mathcal{MN}(0, \Sigma \times I_n)$. We only list the results without repeating similar calculations. For all calculations we assume Y is given.

E-step

The log-likelihood of (X, W) equals to

$$l_c(\mu, \Phi, \Sigma | X, W) = \log P(W|\mu, \Phi) + \log P(X|W, \Sigma)$$
$$= -p \log |\Phi| - tr\big((W - M)^T \Phi^{-1}(W - M)\big) - n \log |\Sigma| - tr\big((X - W)\Sigma^{-1}(X - W)^T\big).$$

The conditional distribution $W|X$ is

$$W|X \sim \mathcal{MN}(\mu_W, \Sigma_W),$$

where

$$\Sigma_W = \Omega_W^{-1} = \big(I_p \otimes \Phi^{-1} + \Sigma^{-1} \otimes I_n\big)^{-1},$$

and

$$vec(\mu_W) = \Sigma_W \Big((I_p \otimes \Phi^{-1}) vec(M) + (\Sigma^{-1} \otimes I_n) vec(X)\Big).$$

We take expectation of $l_c(\mu, \Phi, \Sigma | X, W)$ with respect to $W|X$, and obtain

$$\mathbb{E}_{W|X} l_c(\Phi, \Sigma | X, Z) = -p \log |\Phi| - tr\Big(\Phi^{-1}\big((\mu_W - M)(\mu_W - M)^T + \sum_{j=1}^{p} (\Sigma_W)_{jj}\big)\Big)$$
$$- n \log |\Sigma| - tr\Big(\Sigma^{-1}\big((X - \mu_W)^T (X - \mu_W) + tr_n(\Sigma_W)\big)\Big).$$

M-step

Similarly as in Section A.2.1, we denote $S_1 = \big((X - \mu_W)^T (X - \mu_W) + tr_n(\Sigma_W)\big)/n$ and $S_2 = \big((\mu_W - M)((\mu_W - M)^T + \sum_{j=1}^{p} (\Sigma_W)_{jj}\big)/p$. We can update Σ and Φ by the same formula, but note that the quantities now depend on M (or μ), so we also need to estimate μ. If we take derivative of the log-likelihood of $X|Y$ w.r.t $vec(\mu)$ and set it to zero, we would obtain (2.11). So we could either update μ, Φ, Σ iteratively, e.g., first updating Φ, Σ with estimated μ from last iteration, and update μ based on equation (2.11). Another way is first centering the dataset within each class. Then

we could estimate Φ, Σ by the procedure as described in section A.2.1, and update μ by (2.11) after obtaining $\widehat{\Sigma}$ and $\widehat{\Phi}$.

A.2.3 Efficient Calculation

In this subsection we show how to efficiently calculate the quantities needed in the EM algorithm. The notations used are consistent with the classification setting, where a data matrix $X_{n\times p}$ follows a Kronecker sum matrix variate distribution with a general known mean $M_{n\times p}$, including $M = 0_{n\times p}$ as a special case. Calculations are based on the following properties of Kronecker product operation (when dimensions are compatible):

- $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$;
- $(A \otimes B)^T = A^T \otimes B^T$;
- $(A \otimes B)(C \otimes D) = AC \otimes BD$;
- $(A \otimes B)vec(X) = vec(BXA^T)$;
- $\det(A \otimes B) = (\det A)^n (\det B)^p$.

E-step In E-step, we need three quantities: μ_W, $\sum_{j=1}^{q}(\Sigma_W)_{jj}$ and $tr_n(\Sigma_W)$.

Recall $\Sigma_W = \Omega_W^{-1} = \left(\Sigma^{-1} \otimes I_n + I_p \otimes \Phi^{-1}\right)^{-1}$. Denote $\Sigma = U_1\Lambda_1 U_1^T$ and $\Phi = U_2\Lambda_2 U_2^T$ as their eigen-decomposition, and let $Q_1 = U_1\Lambda_1^{1/2}$, then

$$\begin{aligned}
tr\big((\Sigma_W)_{ij}\big) &= tr\Big(\big(Q_{i\cdot} \otimes U_2\big)\big(I_p \otimes I_n + \Lambda_1 \otimes \Lambda_2^{-1}\big)^{-1}\big((Q_{j\cdot})^T \otimes U_2^T\big)\Big) \\
&= tr\Big(\big((Q_{j\cdot})^T Q_{i\cdot} \otimes I_n\big)\big(I_p \otimes I_n + \Lambda_1 \otimes \Lambda_2^{-1}\big)^{-1}\Big) \\
&= tr\Big((Q_{j\cdot})^T Q_{i\cdot} tr_n\left((I_p \otimes I_n + \Lambda_1 \otimes \Lambda_2^{-1})^{-1}\right)\Big) \\
&= Q_{i\cdot} tr_n\left((I_p \otimes I_n + \Lambda_1 \otimes \Lambda_2^{-1})^{-1}\right)(Q_{j\cdot})^T.
\end{aligned}$$

So

$$tr_n(\Sigma_W) = Qtr_n\left((I_p \otimes I_n + \Lambda_1 \otimes \Lambda_2^{-1})^{-1}\right) Q^T,$$

which is a $p \times p$ matrix. The middle part is easy to calculate.

On the other hand, note the fact that

$$\sum_{j=1}^{p} (\Sigma_W)_{jj} = tr_p(\widetilde{\Sigma}_W)$$

where $\widetilde{\Sigma}_W = \left(\Phi^{-1} \otimes I_p + I_n \otimes \Sigma^{-1}\right)^{-1}$. Therefore,

$$\sum_{j=1}^{p} (\Sigma_W)_{jj} = tr_p(\widetilde{\Sigma}_Z) = \widetilde{Q}tr_p\left((I_n \otimes I_q + \Lambda_2 \otimes \Lambda_1^{-1})^{-1}\right) \widetilde{Q}^T,$$

where $\widetilde{Q} = U_2\Lambda_2^{1/2}$.

For μ_W, we could simplify it as

$$\begin{aligned}
\mu_W &= \Sigma_W\Big((\Sigma^{-1} \otimes I_n)vec(X) + (I_p \otimes \Phi^{-1})vec(M)\Big) \\
&= \left(I_p \otimes I_n + \Sigma \otimes \Phi^{-1}\right)^{-1}\Big(vec(X) + (\Sigma \otimes \Phi^{-1})vec(M)\Big).
\end{aligned}$$

Denote $vec(X) + (\Sigma \otimes \Phi^{-1})vec(Z) = vec(S_1)$, then

$$\begin{aligned}
&\left(I_p \otimes I_n + \Sigma \otimes \Phi^{-1}\right)^{-1} vec(S_1) \\
=&(U_1 \otimes U_2)\left(I_p \otimes I_n + \Lambda_1 \otimes \Lambda_2^{-1}\right)^{-1}(U_1 \otimes U_2)^T vec(S_1) \\
=&(U_1 \otimes U_2)\left(I_p \otimes I_n + \Lambda_1 \otimes \Lambda_2^{-1}\right)^{-1} vec(U_2^T S_1 U_1) \\
=&vec(U_2 S_2 U_1^T)
\end{aligned}$$

where $vec(S_2) = \left(I_p \otimes I_n + \Lambda_1 \otimes \Lambda_2^{-1}\right)^{-1} vec(U_2^T S_1 U_1)$.

Calculation of μ For equation (2.11), we have

$$\begin{aligned}
&\Big((I_p \otimes C)^T(\Sigma \otimes I_n + I_p \otimes \Phi)^{-1}(I_p \otimes C)\Big)^{-1} \\
&= \Big((U_1\Lambda_1^{-1/2} \otimes C^T U_2)\big(I_n \otimes I_p + \Lambda_1^{-1} \otimes \Lambda_2\big)^{-1}(\Lambda_1^{-1/2}U_1^T \otimes U_2^T C)\Big)^{-1},
\end{aligned}$$

and

$$\begin{aligned}
&(I_p \otimes C)^T(I_p \otimes \Phi + \Sigma \otimes I_n)^{-1}vec(X) \\
=&(U_1\Lambda_1^{-1/2} \otimes C^T U_2)\big(I_n \otimes I_p + \Lambda_1^{-1} \otimes \Lambda_2\big)^{-1}(\Lambda_1^{-1/2}U_1^T \otimes U_2^T)vec(X) \\
=&(U_1\Lambda_1^{-1/2} \otimes C^T U_2)\big(I_n \otimes I_p + \Lambda_1^{-1} \otimes \Lambda_2\big)^{-1}vec(U_2^T X U_1\Lambda_1^{-1/2}) \\
=&vec(C^T U_2 S U_1^T A^{-1/2}),
\end{aligned}$$

where $vec(S) = \big(I_n \otimes I_p + \Lambda_1^{-1} \otimes \Lambda_2\big)^{-1}vec(U_2^T X U_1\Lambda_1^{-1/2})$. Multiplying these two parts gives $vec(\hat{\mu})$.

Calculation of likelihood of X If we would like to calculate the likelihood of X and use it as the stopping criterion for the EM algorithm, we need to evaluate $\log\det(I_p \otimes \Phi + \Sigma \otimes I_n)$ and $\big(vec(X) - vec(M)\big)^T(I_p \otimes \Phi + \Sigma \otimes I_n)^{-1}\big(vec(X) - vec(M)\big)$.

One term can be simplied as

$$\begin{aligned}
&\big(vec(X) - vec(M)\big)^T(I_p \otimes \Phi + \Sigma \otimes I_n)^{-1}\big(vec(X) - vec(M)\big) \\
=&\big(vec(X) - vec(M)\big)^T(U_1\Lambda_1^{-1/2} \otimes U_2)(I_p \otimes I_n + \Lambda_1^{-1} \otimes \Lambda_2)^{-1} \\
&(\Lambda_1^{-1/2}U_1^T \otimes U_2^T)\big(vec(X) - vec(M)\big) \\
=&vec(U_2^T(X - M)U_1\Lambda_1^{-1/2})^T(I_p \otimes I_n + \Lambda_1^{-1} \otimes \Lambda_2)^{-1}vec(U_2^T(X - M)\Lambda_1^{-1/2}U_1^T).
\end{aligned}$$

For the other, we have

$$\begin{aligned}
\det\Sigma \otimes I_n + I_p \otimes \Phi &= \det(U_1\Lambda_1^{1/2} \otimes U_2)\det(I_n \otimes I_p + \Lambda_1^{-1} \otimes \Lambda_2)\det(\Lambda_1^{-1/2}U_1^T \otimes U_2^T) \\
&= (\det(\Lambda_1^{1/2})^n)\det(I_n \otimes I_p + \Lambda_1^{-1} \otimes \Lambda_2)(\det(\Lambda_1^{1/2})^n) \\
&= (\det\Lambda_1)^n\det(I_n \otimes I_p + \Lambda_1^{-1} \otimes \Lambda_2),
\end{aligned}$$

so

$$\log\det\Sigma\otimes I_n + I_p\otimes\Phi = n\sum_{i=1}^{p}\log\Lambda_1^i + \sum_{i=1}^{p}\sum_{j=1}^{n}\log(1+\frac{\Lambda_2^j}{\Lambda_1^i}),$$

where Λ_1^i are the ith eigenvalue of Σ so does for Λ_2^j.

Based on the above, we could see that inverse of an $np\times np$ matrix or multiplication of such size matrix could be avoided by taking eigenvalue decomposition of matrix of size $n\times n$ and $p\times p$, and then utilize properties of Kronecker product operations.

A.3 Predicting Class Labels by Variational Methods

In the classification problem, we predict the unobserved Y^* to be the assignment of labels of that maximizes $P(Y^*|X^*)$. However maximizing this quantity directly is intractable so we consider approximate variational methods. The main idea is we pick a family of distribution over the unobserved variable Y^* with its own variational parameters, $q(Y^*|\tau)$ that makes the maximization tractable, and find the setting of parameters that makes q close to the distribution of interest, $P(Y^*|X^*)$.

Here we consider the mean-field approximation approach. That means, we specify

$$q(Y_1^*, Y_2^*, \ldots, Y_n^*) = \prod_{i=1}^{n} q(Y_i^*) = \prod_{i=1}^{n}\prod_{k=1}^{K}\tau_{ik}^{Y_{ik}},$$

with $Y_{ik} = \mathbf{1}(Y_i = k)$ and variational parameters $\tau \coloneqq \{\tau_{ik}\}_{1\le i\le n, 1\le k\le K}$. Under this distribution, $Y_1^*, Y_2^*, \ldots, Y_n^*$ are independently distributed, with

$$Y_i^* \sim \text{Multinomial}(\tau_{i1}, \ldots, \tau_{ik}).$$

We would like to find τ that minimizes the Kullback-Leibler (KL) divergence between $q(Y^*)$ and $P(Y^*|X^*)$, denoted by $\text{KL}(q||p)$. Following the derivation in (*Blei et al.*,

2017), minimizing $\mathrm{KL}(q||p)$ is the same as maximizing the quantity:

$$\mathcal{L} = -\mathbb{E}_q \left(\log P(X^*, Y^*)\right) - \mathbb{E}_q \left(\log q(Y^*)\right).$$

To maximize $\mathcal{L}$ with respect to $q(Y_i^*)$, we could take derivative of this quantity w.r.t to $q(Y_i^*)$ and set it to zero, from which we obtain

$$q(Y_i^*) \propto \exp\left\{\mathbb{E}_{-i}\left[\log P(X^*, Y_i^*, Y_{-i}^*)\right]\right\}.$$

The subscript of the expectation means $q(Y_{-i}^*)$. This implies we could estimate τ by coordinate ascent algorithms, i.e., iteratively updating each τ_{ik} by,

$$\tau_{ik} \propto \pi_k \exp \mathbb{E}_{-i} \left\{ -\frac{1}{2}(vec(X^*) - vec(M)^T \left(\Sigma \otimes I_n + I_p \otimes \Phi\right)^{-1} \left(vec(X^*) - vec(M)\right) \right\},$$

$$\text{subject to } \sum_{k=1}^{K} \tau_{ik} = 1.$$

The part in the exponential could be simplified by only keeping terms involving Y_i^*. We denote $\Omega = (I_n \otimes \Sigma + \Phi \otimes I_p)^{-1}$, the precision matrix of $(X^*)^T$. We view Ω being composed of $n \times n$ blocks, with each block of size $p \times p$. $\Omega_{ii'}$ denotes its (i, i')th block. Then we have

$$\tau_{ik} \propto \pi_k \exp \left\{ -\frac{1}{2}(X_{i\cdot}^* - \mu_k)^T \Omega_{ii} (X_{i\cdot}^* - \mu_k) - \sum_{i' \neq i} (X_{i\cdot}^* - \mu_k)^T \Omega_{ii'} (X_{i'\cdot}^* - m_{i'}) \right\},$$

where $m_{i'} = \mathbb{E}_{-i} M_{i'\cdot} = \mathbb{E}_{-i} \left(\sum_{k=1}^{K} \mathbf{1}(Y_{i'}^* = k) \mu_k \right) = \sum_{k=1}^{K} \tau_{i'k} \mu_k$. Under the distribution $q(Y_1^*, Y_2^*, \ldots, Y_n^*)$, we could predict Y_i^* by $\widehat{Y}_i^* = \arg\max_k \tau_{ik}$.

APPENDIX B

Appendix of Chapter 3

B.1 Proof of Theorem III.5

We consider minimizing the objective function

$$L(Z, \alpha, B, \gamma) = L_A + \lambda L_Y,$$

where L_A and L_Y are defined as in (3.6) and (3.7).

Denote

$$(\widehat{Z}, \widehat{\alpha}, \widehat{B}, \widehat{\gamma}) = \arg\min_{\Theta \in \mathcal{F}} L(Z, \alpha, B, \gamma),$$

and

$$\widehat{\Theta} = \left[\widehat{\Theta}^A, \widehat{\Theta}^Y\right] = \left[\widehat{\alpha} 1_n^T + 1_n^T \widehat{\alpha} + \widehat{Z}\widehat{Z}^T, 1_n \widehat{\gamma} + \widehat{Z}\widehat{B}\right],$$

then

$$L(Z^*, \alpha^*, B^*, \gamma^*) - L(\widehat{Z}, \widehat{\alpha}, \widehat{B}, \widehat{\gamma}) \geq 0. \tag{B.1}$$

Moreover,

$$
\begin{aligned}
&L(Z^*, \alpha^*, B^*, \gamma^*) - L(\widehat{Z}, \widehat{\alpha}, \widehat{B}, \widehat{\gamma}) \\
=&\sum_{i=1}^{n}\sum_{i'=1}^{n}\left\{A_{ii'}(\widehat{\Theta}_{ii'}^{A} - \Theta_{ii'}^{*A}) - (f_A(\widehat{\Theta}_{ii'}^{A}) - f_A(\Theta_{ii'}^{*A})\right\} \\
+&\lambda\sum_{i=1}^{n}\sum_{j=1}^{q}\left\{Y_{ij}(\widehat{\Theta}_{ij}^{Y} - \Theta_{ij}^{*Y}) - (f_Y(\widehat{\Theta}_{ij}^{Y}) - f_Y(\Theta_{ij}^{*Y})\right\}.
\end{aligned}
$$

Using Taylor's expansion, we have the last equation equal to

$$
\begin{aligned}
&\sum_{i=1}^{n}\sum_{i'=1}^{n}\left\{\left(A_{ii'} f_A'(\Theta_{ii'}^{*A})\right)(\widehat{\Theta}_{ii'}^{A} - \Theta_{ii'}^{*A})\right\} \\
-&\sum_{i=1}^{n}\sum_{i'=1}^{n}\left\{f_A(\widehat{\Theta}_{ii'}^{A}) - f_A(\Theta_{ii'}^{*A}) - f_A'(\Theta_{ii'}^{*A})(\widehat{\Theta}_{ii'}^{A} - \Theta_{ii'}^{*A})\right\} \\
+&\lambda\sum_{i=1}^{n}\sum_{j=1}^{q}\left\{\left(Y_{ij} - f_Y'(\Theta_{ij}^{*Y})\right)(\widehat{\Theta}_{ij}^{Y} - \Theta_{ij}^{*Y})\right\} \\
-&\lambda\sum_{i=1}^{n}\sum_{j=1}^{q}\left\{f_Y(\widehat{\Theta}_{ij}^{Y}) - f_Y(\Theta_{ij}^{*Y}) - f_Y'(\Theta_{ij}^{*Y})(\widehat{\Theta}_{ij}^{Y} - \Theta_{ij}^{*Y})\right\} \\
=&\sum_{i=1}^{n}\sum_{i'=1}^{n}\left\{\left(A_{ii'} - f_A'(\Theta_{ii'}^{*A})\right)(\widehat{\Theta}_{ii'}^{A} - \Theta_{ii'}^{A})\right\} - \sum_{i=1}^{n}\sum_{i'=1}^{n}\frac{1}{2}f_A''(\widetilde{\Theta}_{ii'}^{A})(\widehat{\Theta}_{ii'}^{A} - \Theta_{ii'}^{*A})^2 \\
+&\lambda\sum_{i=1}^{n}\sum_{j=1}^{q}\left\{\left(Y_{ij} - f_Y'(\Theta_{ij}^{*Y})\right)(\widehat{\Theta}_{ij}^{Y} - \Theta_{ij}^{*Y})\right\} - \lambda\sum_{i=1}^{n}\sum_{j=1}^{q}\frac{1}{2}f_Y''(\widetilde{\Theta}_{ij}^{Y})(\widehat{\Theta}_{ij}^{Y} - \Theta_{ij}^{*Y})^2,
\end{aligned}
$$

where $\widetilde{\Theta}_{ii'}^{A} = a\Theta_{ii'}^{*A} + (1-a)\widehat{\Theta}_{ii'}^{A}$ for some $a \in (0,1)$ and $\widetilde{\Theta}_{ij'}^{Y} = b\Theta_{ij}^{*Y} + (1-b)\widehat{\Theta}_{ij}^{A}$ for some $b \in (0,1)$. Since we require that both $\widehat{\Theta}$ and Θ^* belong to the feasible parameter space as defined in (3.10), we have the above equation be bounded by

$$
\begin{aligned}
&\sum_{i=1}^{n}\sum_{i'=1}^{n}\left\{\left(A_{ii'} - f_A'(\Theta_{ii'}^{*A})\right)(\widehat{\Theta}_{ii'}^{A} - \Theta_{ii'}^{*A})\right\} \\
+&\lambda\sum_{i=1}^{n}\sum_{j=1}^{q}\left\{\left(Y_{ij} - f_Y'(\Theta_{ij}^{*Y})\right)(\widehat{\Theta}_{ij}^{Y} - \Theta_{ij}^{*Y})\right\} \\
-&\frac{1}{2}\min_{|v|<M_1} f_A''(v)\|\widehat{\Theta}_A - \Theta^{*A}\|_F^2 - \frac{1}{2}\lambda\min_{|v|<M_2} f_Y''(v)\|\widehat{\Theta}_Y - \Theta^{*Y}\|_F^2.
\end{aligned}
$$

Based on (B.1) and the above calculation, we have

$$
\begin{aligned}
&\frac{1}{2}\min_{|v|<M_1} f_A''(v)\|\widehat{\Theta}^A-\Theta^{*A}\|_F^2+\frac{1}{2}\lambda\min_{|v|<M_2} f_Y''(v)\|\widehat{\Theta}^Y-\Theta^{*Y}\|_F^2\\
\leq&\sum_{i=1}^{n}\sum_{i'=1}^{n}\left\{\left(A_{ii'}-f_A'(\Theta_{ii'}^{*A})\right)\left(\widehat{\Theta}_{ii'}^A-\Theta_{ii'}^{*A}\right)\right\}+\lambda\sum_{i=1}^{n}\sum_{j=1}^{q}\left\{\left(Y_{ij}-f_Y'(\Theta_{ij}^{*Y})\right)\left(\widehat{\Theta}_{ij}^Y-\Theta_{ij}^{*Y}\right)\right\}\\
=&\left\langle Z^A,\widehat{\Theta}^A-\Theta^{*A}\right\rangle+\lambda\left\langle Z^Y,\widehat{\Theta}^Y-\Theta^{*Y}\right\rangle\\
\leq&\max(\lambda,1)\left|\left\langle Z,\widehat{\Theta}-\Theta^*\right\rangle\right|.
\end{aligned}
$$

Moreover,

$$
\begin{aligned}
&\min_{|v|<M_1} f_A''(v)\|\widehat{\Theta}^A-\Theta^{*A}\|_F^2+\lambda\min_{|v|<M_2} f_Y''(v)\|\widehat{\Theta}^Y-\Theta^{*Y}\|_F^2\\
\geq&\min(\min_{|v|<M_1} f_A''(v),\lambda\min_{|v|<M_2} f_Y''(v))\left(\|\widehat{\Theta}^A-\Theta^{*A}\|_F^2+|\widehat{\Theta}^Y-\Theta^{*Y}\|_F^2\right)\\
=&\min(\min_{|v|<M_1} f_A''(v),\lambda\min_{|v|<M_2} f_Y''(v))\|\widehat{\Theta}-\Theta^*\|_F^2.
\end{aligned}
$$

Note that $\left|\left\langle Z,\widehat{\Theta}-\Theta^*\right\rangle\right|\leq\|Z\|_{op}\sqrt{\text{rank}(\widehat{\Theta}-\Theta^*)}\|\widehat{\Theta}-\Theta^*\|_F$, where $\|\cdot\|_{op}$ is defined as the operator norm, i.e., the largest singular value of a matrix. Also, $\text{rank}(\widehat{\Theta}-\Theta^*)\leq 2(k+2+k+1)$, therefore, we have

$$
\|\widehat{\Theta}-\Theta^*\|_F\leq\frac{2\max(\lambda,1)\sqrt{2(2k+3)}}{\min(\min_{|v|<M_1} f_A''(v),\lambda\min_{|v|<M_2} f_Y''(v))}\|Z\|_{op}
$$

Similar as the proof in *Chen et al.* (2019b), we utilize the following lemma to bound $\|Z\|_{op}$.

Lemma B.1 (Theorem 2 in *Latała* (2005))**.** *For any finite matrix $Z=\{Z_{ij}\}\in\mathbb{R}^{n\times m}$ of independent mean zero random variables, there exists a constant κ_0 such that*

$$
\mathbb{E}\|Z\|_{op}\leq\kappa_0\left[\max_i\left(\sum_j\mathbb{E}(Z_{ij}^2)\right)^{1/2}+\left(\max_j\sum_i\mathbb{E}(Z_{ij}^2)\right)^{1/2}+\left(\sum_{ij}\mathbb{E}(Z_{ij}^2)\right)^{1/4}\right].
$$

In our case, $Z = [A - f'_A(\Theta^{*A}), Y - f'_Y(\Theta^{*Y})] \in \mathbb{R}^{n\times(n+q)}$, with entries $\{Z_{ij}\}$, $1 \le i \le n, 1 \le j \le (n+q)$, follows some type of generalized linear model. Since entries in Θ^{*A} and Θ^{*Y} are bounded, then we can see that both the second moment and forth moment of $\{Z_{ij}\}_{1\le i\le n, 1\le j\le(n+q)}$ are bounded and there exists an absolute constant κ_1 such that

$$\mathbb{E}(Z_{ij}^2) \le \kappa_1^2, \text{ and } \mathbb{E}(Z_{ij}^4) \le \kappa_1^4, \ i = 1, \ldots, n, j = 1, \ldots, n+q.$$

By Lemma B.1, we have

$$\mathbb{E}\|Z\|_{op} \le \kappa_0\kappa_1 \left(\sqrt{n} + \sqrt{n+q} + \sqrt[4]{n(n+q)}\right) \le 3\kappa_0\kappa_1\sqrt{n+q}.$$

Therefore,

$$\mathbb{E}\|\widehat{\Theta} - \Theta^*\|_F \le \frac{\kappa \max(\lambda, 1)\sqrt{(2k+3)}}{\min\left(\min_{|v|<M_1} f''_A(v), \lambda \min_{|v|<M_2} f''_Y(v)\right)}\sqrt{n+q}.$$

for some constant absolute κ.

The second part of Theorem III.5 is a direct result of Theorem 3 in *Chen et al.* (2019b). It states that there exists $\epsilon'_1 > 0$ and n_0, q_0 such that when $n \ge n_0$, $q \ge q_0$, there exists $\Theta^{0,Y}$ such that

$$P\left(\frac{1}{\sqrt{nq}}\|\overline{\Theta}^Y - \Theta^{0,Y}\|_F \ge \frac{\epsilon'_1}{\sqrt{\min(n,q)}}\right) \ge \frac{1}{2}$$

for arbitrary estimator $\overline{\Theta}^Y \in \mathbb{R}^{n\times q}$. For any estimator $\overline{\Theta} \in \mathbb{R}^{n\times(n+q)}$, we have

$$\|\overline{\Theta} - \Theta^0\|_F \ge \|\overline{\Theta}^Y - \Theta^{0,Y}\|_F.$$

Since $q = \mathcal{O}(n)$, then there exists a constant $\overline{C}$ such that $q \le \overline{C}n$. Taking $\epsilon_1 =$

$\epsilon_1'/\sqrt{(1+\overline{C})}$, we have

$$
\begin{aligned}
& P\left(\frac{1}{\sqrt{n(n+q)}}\|\overline{\Theta}-\Theta^0\|_F \geq \frac{\epsilon_1}{\sqrt{n}}\right) \\
\geq & P\left(\frac{1}{\sqrt{n(n+q)}}\|\overline{\Theta}-\Theta^0\|_F \geq \frac{\epsilon_1'}{\sqrt{n+q}}\right) \\
\geq & P\left(\frac{1}{\sqrt{n(n+q)}}\|\overline{\Theta}^Y-\Theta^{0,Y}\|_F \geq \frac{\epsilon_1'}{\sqrt{n+q}}\right) \\
= & P\left(\frac{1}{\sqrt{nq}}\|\overline{\Theta}^Y-\Theta^{0,Y}\|_F \geq \frac{\sqrt{n(n+q)}}{\sqrt{nq}}\frac{\epsilon_1'}{\sqrt{n+q}}\right) \\
\geq & P\left(\frac{1}{\sqrt{nq}}\|\overline{\Theta}^Y-\Theta^{0,Y}\|_F \geq \frac{\epsilon_1'}{\sqrt{\min(n,q)}}\right) \geq \frac{1}{2}.
\end{aligned}
$$

B.2 Proof of Theorem III.9 and Proposition III.10

In this section we derive the results of Theorem III.9 and Proposition III.10. The calculation is based on continuous Y case, and when Y follows other generalized linear models, the calculation is similar and the proof is omitted here. Also, for continuous Y, we assume that Y has been centered and $\gamma = 0$ for simplicity. We start with proving Propsition III.10, as it deals with a more general case regarding $\widetilde{Z}$, $\widetilde{\alpha}$, $\widetilde{B}$ and $\widetilde{\gamma}$. Then we show the results for Theorem III.9, under a special case of $\widetilde{Z}$ and $\widetilde{\alpha}$.

B.2.1 Proof of Proposition III.10

When Y are continuous, the objective function can be written as

$$
\begin{aligned}
&L(Z,\alpha,B) = -\sum_{i,i'}\{A_{ii'}\Theta_{ii'} + \log\left(1-\sigma(\Theta_{ii'})\right)\} + \frac{\lambda}{2}\|Y-ZB\|_F^2 \\
=&(\lambda+2)\left[-\frac{2}{\lambda+2}\times\frac{1}{2}\sum_{i,i'}\{A_{ii'}\Theta_{ii'} + \log\left(1-\sigma(\Theta_{ii'})\right)\} + \frac{\lambda}{\lambda+2}\times\frac{1}{2}\|Y-ZB\|_F^2\right] \\
=&(\lambda+2)\left[(1-\widetilde{\lambda})\times-\frac{1}{2}\sum_{i,i'}\{A_{ii'}\Theta_{ii'} + \log\left(1-\sigma(\Theta_{ii'})\right)\} + \widetilde{\lambda}\times\frac{1}{2}\|Y-ZB\|_F^2\right],
\end{aligned}
$$

where $\widetilde{\lambda} = \dfrac{\lambda}{\lambda+2}$. Denote

$$
\widetilde{L}(Z,\alpha,B) = (1-\widetilde{\lambda})\times-\frac{1}{2}\sum_{i,i'}\{A_{ii'}\Theta_{ii'} + \log\left(1-\sigma(\Theta_{ii'})\right)\} + \widetilde{\lambda}\times\frac{1}{2}\|Y-ZB\|_F^2.
$$

Then minimizing $L(Z,\alpha,B)$ is equivalent to minimizing $\widetilde{L}(Z,\alpha,B)$. Suppose we update Z by gradient descent method as in Algorithm 3, and at each iteration, Z^t is updated by

$$
Z^{t+1} = Z^t - \eta_z\nabla_Z\widetilde{L}(Z,\alpha,B) = Z^t + (1-\widetilde{\lambda})\eta_z(A-\sigma(\Theta^t))Z^t + \widetilde{\lambda}\eta_z\left(Y-Z^tB^t\right)\left(B^t\right)^T. \tag{B.2}
$$

The updated Z^{t+1} is based on the estimated paramaters Z^t, α^t and B^t from last iteration. To demonstrate the results of such one-step update, we consider the case that we are given fixed initial values of parameters Z, α, and B, denoted by $\widetilde{Z}$, $\widetilde{\alpha}$, and $\widetilde{B}$ respectively. Suppose these initial estimates satisfy the conditions that $\|\widetilde{Z}-Z\|_F^2 = \mathcal{O}(1)$, $\|\widetilde{\alpha}1_n^T - \alpha 1_n^T\|_F^2 = \mathcal{O}(n)$, and $\|\widetilde{B}-B\|_F^2 = \mathcal{O}(1)$. In other words, the initial estimates are close to the true parameters. Given $\widetilde{Z}$, $\widetilde{\alpha}$, and $\widetilde{B}$, we consider

update $\widetilde{Z}$ one step by (B.2) and obtain a $\widehat{Z}$, i.e.,

$$\begin{aligned}\widehat{Z} &= \widetilde{Z} + (1-\widetilde{\lambda})\eta_z(A-\sigma(\widetilde{\Theta}))\widetilde{Z} + \widetilde{\lambda}\eta_z\left(Y-\widetilde{Z}\widetilde{B}\right)\widetilde{B}^T \\ &= \widetilde{Z} + (1-\widetilde{\lambda})\rho_1 n^{-1}(A-\sigma(\widetilde{\Theta}))\widetilde{Z}^T + \widetilde{\lambda}\rho_2 q^{-1}\left(Y-\widetilde{Z}\widetilde{B}\right)\widetilde{B}^T,\end{aligned} \tag{B.3}$$

where $\rho_1 = \eta_z n = \mathcal{O}(1)$, and $\rho_2 = \eta_z q = \mathcal{O}(1)$.

For a particular node i, we have

$$\widehat{Z}_i = \widetilde{Z}_i + \widetilde{\lambda}\rho_2 q^{-1}\widetilde{B}(Y_i - \widetilde{B}^T\widetilde{Z}_i) + (1-\widetilde{\lambda})\rho_1 n^{-1}\widetilde{Z}^T(A_{i\cdot}^T - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}\widetilde{Z}_i)).$$

Here, $\widetilde{\boldsymbol{\alpha}}_i$ denote the ith column of the matrix $\widetilde{\alpha}1_n^T + 1_n\widetilde{\alpha}^T$, i.e., $\widetilde{\boldsymbol{\alpha}}_i = (\widetilde{\alpha}_i+\widetilde{\alpha}_1, \ldots, \widetilde{\alpha}_i+\widetilde{\alpha}_n)$. Correspondingly, we denote $\boldsymbol{\alpha}_i$ as the ith column of the matrix $\alpha 1_n^T + 1_n\alpha^T$ in the following calculations.

Then

$$\begin{aligned}&\widehat{Z}_i - Z_i \\ =&\widetilde{Z}_i - Z_i + \widetilde{\lambda}\rho_2 q^{-1}\widetilde{B}(Y_i - \widetilde{B}^T\widetilde{Z}_i) + (1-\widetilde{\lambda})\rho_1 n^{-1}\widetilde{Z}^T(A_{i\cdot}^T - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}\widetilde{Z}_i)) \\ =&\widetilde{Z}_i - Z_i + \widetilde{\lambda}\rho_2 q^{-1}\widetilde{B}(Y_i - B^TZ_i + B^TZ_i - \widetilde{B}^TZ_i + \widetilde{B}^TZ_i - \widetilde{B}^T\widetilde{Z}_i) + (1-\widetilde{\lambda})\rho_1 n^{-1}\widetilde{Z}^T \\ &\left(A_{i\cdot}^T - \sigma(\boldsymbol{\alpha}_i + ZZ_i) + \sigma(\boldsymbol{\alpha}_i + ZZ_i) - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}Z_i) + \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}Z_i) - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}\widetilde{Z}_i)\right) \\ =&\widetilde{\lambda}\left\{\left(I_k - \rho_2 q^{-1}\widetilde{B}\widetilde{B}^T\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_2 q^{-1}\widetilde{B}(B-\widetilde{B})^TZ_i\right\} + (1-\widetilde{\lambda}) \\ &\left\{\left(I_k - \rho_1 n^{-1}\widetilde{Z}^T diag(\sigma'(\xi_i))\widetilde{Z}\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_1 n^{-1}\widetilde{Z}^T\left(\sigma(\boldsymbol{\alpha}_i + ZZ_i) - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}Z_i)\right)\right\} \\ &+\widetilde{\lambda}\rho_2 q^{-1}\widetilde{B}E_i + (1-\widetilde{\lambda})\rho_1 n^{-1}\widetilde{Z}^T\left(A_{i\cdot}^T - \sigma(\boldsymbol{\alpha}_i + ZZ_i)\right).\end{aligned}$$

Here ξ_i is a vector between $\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}Z_i$ and $\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}\widetilde{Z}_i$.

For notational simplicity, we denote

$$\begin{aligned}T_{i1} = &\widetilde{\lambda}\left\{\left(I_k - \rho_2 q^{-1}\widetilde{B}\widetilde{B}^T\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_2 q^{-1}\widetilde{B}(B-\widetilde{B})^TZ_i\right\} + (1-\widetilde{\lambda}) \\ &\left\{\left(I_k - \rho_1 n^{-1}\widetilde{Z}^T diag(\sigma'(\xi_i))\widetilde{Z}\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_1 n^{-1}\widetilde{Z}^T\left(\sigma(\boldsymbol{\alpha}_i + ZZ_i) - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}Z_i)\right)\right\}\end{aligned}$$

and

$$T_{i2} = \widetilde{\lambda}\rho_2 q^{-1}\widetilde{B}E_i + (1-\widetilde{\lambda})\rho_1 n^{-1}\widetilde{Z}^T\left(A_{i\cdot}^T - \sigma(\boldsymbol{\alpha}_i + ZZ_i)\right).$$

Then we have

$$\mathbb{E}\|\widehat{Z}_i - Z_i\|^2 = T_{i1}^2 + \mathbb{E}T_{i2}^2 + \mathbb{E}\langle T_{i1}, T_{i2}\rangle.$$

The inner-product term equals to 0 since T_{i2} is a mean 0 random vector. Moreover, since in our model assumption, Y and A are conditionally independent given Z, then

$$\mathbb{E}T_{i2}^2 = \widetilde{\lambda}^2\rho_2^2 q^{-2}\sigma^2 tr(\widetilde{B}\widetilde{B}^T) + (1-\widetilde{\lambda})^2\rho_1^2 n^{-2} tr(\widetilde{Z}^T W_i \widetilde{Z}) \coloneqq \widetilde{\lambda}^2\widetilde{e}_{iY} + (1-\widetilde{\lambda})^2\widetilde{e}_{iA},$$

where $0 \preceq W_i = diag(\sigma'(\boldsymbol{\alpha}_i + ZZ_i)) \preceq \frac{1}{4}I_n$. We have assumed that$\|\widetilde{B} - B\|_F^2 = \mathcal{O}(1)$. Note that $\widetilde{B}\widetilde{B}^T - BB^T = (\widetilde{B} - B)(\widetilde{B} + B)^T$. Then $\|\widetilde{B}\widetilde{B}^T - BB^T\|_F^2 \leq \|\widetilde{B} - B\|_F^2\|\widetilde{B} + B\|_F^2 = \mathcal{O}(q)$. If we denote the eigenvalues of $\widetilde{B}\widetilde{B}^T/q$ as $(\widetilde{\sigma}_1, \ldots, \widetilde{\sigma}_k)$, then by Weyl's inequality, we have

$$|\widetilde{\sigma}_j - \sigma_j| < \left\|\widetilde{B}\widetilde{B}^T/q - BB^T/q\right\|_2 \leq \left\|\widetilde{B}\widetilde{B}^T/q - BB^T/q\right\|_F = \mathcal{O}\left(1/\sqrt{q}\right).$$

Therefore, $tr(\widetilde{B}\widetilde{B}^T)/q = tr(BB^T)/q + o(1) = \mathcal{O}(1)$. Similarly, denote the eigenvalues of $\widetilde{Z}^T\widetilde{Z}/n$ as $(\widetilde{\lambda}_1, \ldots, \widetilde{\lambda}_k)$. By similar calculation we have $tr(\widetilde{Z}^T W_i \widetilde{Z})/n = \mathcal{O}(1)$. Overally speaking, the term $\mathbb{E}T_{i2}^2$ is dominated by the parameters in the model, but not the estimation error induced by the initial estimates $\widetilde{Z}$, $\widetilde{\alpha}$, and $\widetilde{B}$.

Next, we denote

$$\widetilde{T}_{iY} \coloneqq \left\{\left(I_k - \rho_2 q^{-1}\widetilde{B}\widetilde{B}^T\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_2 q^{-1}\widetilde{B}(B - \widetilde{B})^T Z_i\right\}^T$$
$$\left\{\left(I_k - \rho_2 q^{-1}\widetilde{B}\widetilde{B}^T\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_2 q^{-1}\widetilde{B}(B - \widetilde{B}^T) Z_i\right\}.$$

Correspondingly, we denote

$$\widetilde{T}_{iA} \coloneqq \left\{\left(I_k - \rho_1 n^{-1}\widetilde{Z}^T diag(\sigma'(\xi_i))\widetilde{Z}\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_1 n^{-1}\widetilde{Z}^T\left(\sigma(\boldsymbol{\alpha}_i + ZZ_i) - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}Z_i)\right)\right\}^T$$
$$\left\{\left(I_k - \rho_1 n^{-1}\widetilde{Z}^T diag(\sigma'(\xi_i))\widetilde{Z}\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_1 n^{-1}\widetilde{Z}^T\left(\sigma(\boldsymbol{\alpha}_i + ZZ_i) - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}Z_i)\right)\right\}$$

and

$$\widetilde{T}_{iAY} \coloneqq \left\{\left(I_k - \rho_2 q^{-1}\widetilde{B}\widetilde{B}^T\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_2 q^{-1}\widetilde{B}(B - \widetilde{B})^T Z_i\right\}^T$$
$$\left\{\left(I_k - \rho_2 q^{-1}\widetilde{Z}^T diag(\sigma'(\xi_i))\widetilde{Z}\right)\left(\widetilde{Z}_i - Z_i\right) + \rho_1 n^{-1}\widetilde{Z}^T\left(\sigma(\boldsymbol{\alpha}_i + ZZ_i) - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z}Z_i)\right)\right\}.$$

Therefore, we have

$$\widetilde{T}_{i1}^2 = \widetilde{\lambda}^2\widetilde{T}_{iY} + (1-\widetilde{\lambda})^2\widetilde{T}_{iA} + 2\widetilde{\lambda}(1-\widetilde{\lambda})\widetilde{T}_{iAY}.$$

The mean square error of $\widehat{Z}_i$ can be expressed as

$$\begin{aligned}\mathbb{E}\|\widehat{Z}_i - Z_i\|^2 &= \widetilde{\lambda}^2\widetilde{T}_{iY} + (1-\widetilde{\lambda})^2\widetilde{T}_{iA} + 2\widetilde{\lambda}(1-\widetilde{\lambda})\widetilde{T}_{iAY} + \widetilde{\lambda}^2\widetilde{e}_Y + (1-\widetilde{\lambda})^2\widetilde{e}_{iA} \\ &= \widetilde{\lambda}^2(T_{iY} + T_{iA} - 2T_{iAY} + e_Y + e_{iA}) - 2\widetilde{\lambda}(T_{iA} - T_{iAY} + e_{iA}) + T_{iA} + e_{iA}.\end{aligned} \tag{B.4}$$

Since in each step, $\widetilde{Z}$ are updated for all n data points simutaneously, we can add the results in (B.4) from $i = 1$ to n and obtain

$$\begin{aligned}&\mathbb{E}\|\widehat{Z} - Z\|_F^2 = \sum_{i=1}^n \mathbb{E}\|\widehat{Z}_i - Z_i\|^2 \\ &= \widetilde{\lambda}^2\sum_{i=1}^n(T_{iY} + T_{iA} - 2T_{iAY} + e_Y + e_{iA}) - 2\widetilde{\lambda}\sum_{i=1}^n(T_{iA} - T_{iAY} + e_{iA}) + \sum_{i=1}^n(T_{iA} + e_{iA}).\end{aligned} \tag{B.5}$$

Taking derivative w.r.t $\widetilde{\lambda}$, we have

$$\widetilde{\lambda}_{opt} = \frac{\sum_{i=1}^n(T_{iA} - T_{iAY} + e_{iA})}{\sum_{i=1}^n(T_{iY} + T_{iA} - 2T_{iAY} + e_Y + e_{iA})}$$

that minimizes $\mathbb{E}\|\widehat{Z} - Z\|_F^2$.

We can further simplify and analyze the expression of $\widetilde{\lambda}_{opt}$. First, we denote

$$\begin{aligned}
\widetilde{T}_Y \coloneqq \sum_{i=1}^{n} \widetilde{T}_{iY} &= \sum_{i=1}^{n} \left\| \left(I_k - \rho_2 q^{-1} \widetilde{B}\widetilde{B}^T \right) \left(\widetilde{Z}_i - Z_i \right) + \rho_2 q^{-1} \widetilde{B}(B - \widetilde{B})^T Z_i \right\|_2^2 \\
&= \left\| (\widetilde{Z} - Z) \left(I_k - \rho_2 q^{-1} \widetilde{B}\widetilde{B}^T \right) + \rho_2 q^{-1} \widetilde{B}(B - \widetilde{B})^T Z \right\|_F^2 \\
&= tr\left((\widetilde{Z} - Z) \left(I_k - \rho_2 q^{-1} \widetilde{B}\widetilde{B}^T \right) \left(I_k - \rho q^{-1} \widetilde{B}\widetilde{B}^T \right) (\widetilde{Z} - Z)^T \right) \\
&+ \rho_2^2 q^{-2} tr\left(\widetilde{Z}(B - \widetilde{B})\widetilde{B}^T \widetilde{B}(B - \widetilde{B})^T \widetilde{Z}^T \right) \\
&+ 2\rho_2 \left\langle (\widetilde{Z} - Z) \left(I_k - \rho_2 q^{-1} \widetilde{B}\widetilde{B}^T \right), q^{-1} \widetilde{Z}(B - \widetilde{B})\widetilde{B}^T \right\rangle.
\end{aligned}$$

We calculate each term separately. The first term can be bounded by

$$\begin{aligned}
&tr\left((\widetilde{Z} - Z) \left(I_k - \rho_2 q^{-1} \widetilde{B}\widetilde{B}^T \right) \left(I_k - \rho_2 q^{-1} \widetilde{B}\widetilde{B}^T \right) (\widetilde{Z} - Z)^T \right) \\
\leq & k^2 (1 - \rho\widetilde{\sigma}_k)^2 \|\widetilde{Z} - Z\|_F^2 = \mathcal{O}(1),
\end{aligned} \tag{B.6}$$

and

$$\begin{aligned}
&tr\left(\widetilde{Z}(B - \widetilde{B})\widetilde{B}^T \widetilde{B}(B - \widetilde{B})^T \widetilde{Z}^T \right) \\
\leq & q\widetilde{\sigma}_1 tr\left(\widetilde{Z}(B - \widetilde{B})(B - \widetilde{B})^T \widetilde{Z}^T \right) \\
\leq & q\widetilde{\sigma}_1 \|B - \widetilde{B}\|_2^2 \|\widetilde{Z}\|_F^2 \leq qn\widetilde{\sigma}_1 \widetilde{\lambda}_1 \|B - \widetilde{B}\|_2^2.
\end{aligned} \tag{B.7}$$

So the second term is also of $\mathcal{O}\left(\frac{q}{n}\right) = \mathcal{O}(1)$. The last inner product term is bounded by the norm of each term so is also of $\mathcal{O}(1)$.

Further, we denote the matrix $D_A = diag([\xi_1, \xi_2, \ldots, \xi_n])$, which is a diagonal matrix of size $n^2 \times n^2$, and we denote the matrix $M_A = I_n \otimes I_k - \rho n^{-1}(I_n \otimes \widetilde{Z}^T) D_A (I_n \otimes$

$\widetilde{Z}$). Then we have

$$
\begin{aligned}
&\widetilde{T}_A \coloneqq \sum_{i=1}^{n} \widetilde{T}_{iA} \\
&= \sum_{i=1}^{n} \left\| \left(I_k - \rho_1 n^{-1} \widetilde{Z}^T diag(\sigma'(\xi_i)) \widetilde{Z} \right) \left(\widetilde{Z}_i - Z_i \right) + \rho_1 n^{-1} \widetilde{Z}^T \left(\sigma(\boldsymbol{\alpha}_i + ZZ_i) - \sigma(\widetilde{\boldsymbol{\alpha}}_i + \widetilde{Z} Z_i) \right) \right\|_2^2 \\
&= \left\| vec(\widetilde{Z} - Z) M_A + \rho n^{-1} vec \left\{ \left(\sigma(\boldsymbol{\alpha} + ZZ^T) - \sigma(\widetilde{\boldsymbol{\alpha}} + Z\widetilde{Z}) \right) \widetilde{Z} \right\} \right\|_F^2 \\
&= \left\| vec(\widetilde{Z} - Z) M_A \right\|_F^2 + \rho_1^2 n^{-2} \left\| vec \left\{ \left(\sigma(\boldsymbol{\alpha} + ZZ^T) - \sigma(\widetilde{\boldsymbol{\alpha}} + Z\widetilde{Z}) \right) \widetilde{Z} \right\} \right\|_F^2 \\
&+ 2\rho_1 \left\langle vec(\widetilde{Z} - Z) M_A, n^{-1} vec \left\{ \left(\sigma(\boldsymbol{\alpha} + ZZ^T) - \sigma(\widetilde{\boldsymbol{\alpha}} + Z\widetilde{Z}) \right) \widetilde{Z} \right\} \right\rangle,
\end{aligned}
$$

where $vec(\cdot)$ means stacking the rows of a matrix into a row vector. Note that

$$
\|vec(\widetilde{Z} - Z) M_A\| \leq k^2 (1 - C_A \widetilde{\lambda}_k)^2 \|\widetilde{Z} - Z\|_F^2 = \mathcal{O}(1),
$$

here $C_A \approx \sigma'(M_1) = \sigma(M_1)(1 - \sigma(M_1)) > 0$ with M_1 speficied in Assumption III.3. When the network is sparser, i.e., a lot entries in Θ^A have connecting probabilities close to 0, the M_1 specified in Assumption III.3 would also become larger. Therefore, sparser network leads to smaller C_A. Moreover,

$$
\begin{aligned}
&\|vec((\sigma(\boldsymbol{\alpha} + ZZ^T) - \sigma(\widetilde{\boldsymbol{\alpha}} + Z\widetilde{Z})\widetilde{Z}))\|_F^2 \leq \|\widetilde{Z}\|_F^2 \|\sigma(\boldsymbol{\alpha} + ZZ^T) - \sigma(\widetilde{\boldsymbol{\alpha}} + Z\widetilde{Z}^T)\|_F^2 \\
\leq &\frac{1}{16} \|\widetilde{Z}\|_F^2 \|\alpha + ZZ^T - \widetilde{\alpha} - Z\widetilde{Z}^T\|_F^2 = \mathcal{O}(n^2).
\end{aligned}
$$

So the second term is of $\mathcal{O}(1)$. The inner product term is also bounded by $\mathcal{O}(1)$ by similar argument when calculating $\widetilde{T}_Y$.

The other terms could be bounded as follows:

$$
\begin{aligned}
&\widetilde{T}_{AY} \coloneqq \sum_{i=1}^{n} \widetilde{T}_{iAY} \\
&= vec\left\{(\widetilde{Z}-Z)\left(I_k - \rho_2 q^{-1}\widetilde{B}\widetilde{B}^T\right) + \rho_2 q^{-1}\widetilde{Z}(B-\widetilde{B})\widetilde{B}^T\right\} \\
&\times \left[vec(\widetilde{Z}-Z)M_A + \rho_1 n^{-1} vec\left\{\left(\sigma(\alpha 1_n^T + 1_n\alpha^T + ZZ^T) - \sigma(\widetilde{\alpha}1_n^T + 1_n\widetilde{\alpha} + Z\widetilde{Z})\right)\widetilde{Z}\right\}\right]^T \\
&\leq k^2(1-\rho_1 c_A\widetilde{\lambda}_k)(1-\rho_2\widetilde{\sigma}_k)\|\widetilde{Z}-Z\|_F^2 + \mathcal{O}(1), \\
&\widetilde{e}_Y \coloneqq \sum_{i=1}^{n}\widetilde{e}_{iY} = \sum_{i=1}^{n}\rho_2^2 q^{-2}\sigma^2 tr(\widetilde{B}\widetilde{B}^T) = \rho_2^2 n q^{-2}\sigma^2 tr(\widetilde{B}\widetilde{B}^T) \leq \rho_2^2 n q^{-1}\sigma^2\widetilde{\sigma}_1, \\
&\widetilde{e}_A \coloneqq \sum_{i=1}^{n}\widetilde{e}_{iA} = \sum_{i=1}^{n}\rho_1^2 n^{-2} tr(\widetilde{Z}^T W_i \widetilde{Z}) \\
&= \rho_1^2 n^{-2} tr((I_n\otimes\widetilde{Z})^T diag(vec(\sigma'(\Theta)))(I_n\otimes\widetilde{Z})) \leq \rho_1^2/4\widetilde{\lambda}_1.
\end{aligned}
\tag{B.8}
$$

We can obtain the optimal $\widetilde{\lambda}$ as

$$
\widetilde{\lambda}_{opt} = \frac{\widetilde{T}_A - \widetilde{T}_{AY} + \widetilde{e}_A}{\widetilde{T}_Y + \widetilde{T}_A - 2\widetilde{T}_{AY} + \widetilde{e}_A + \widetilde{e}_Y}.
$$

The upper bounds on T_A and T_{AY} suggests that when C_A is smaller, the upper bounds on T_A would be larger than that of T_{AY} therefore we are more likely to obtain a positive $\widetilde{T}_A - \widetilde{T}_{AY} + \widetilde{e}_A$. This implies that when the information from the network is limited, a positive $\widetilde{\lambda}_{opt}$ would be selected and incorporating nodal variables is preferred.

The above calculation is about choosing an optimal $\widetilde{\lambda}$. For a fixed $\widetilde{\lambda} > 0$, we have

$$
\mathbb{E}\|\widehat{Z}-Z\|_F^2 = \widetilde{\lambda}^2(T_Y + T_A - 2T_{AY} + e_Y + e_A) - 2\widetilde{\lambda}(T_A - T_{AY} + e_A) + (T_A + e_A).
$$

When $\widetilde{\lambda} = 0$, i.e., obtain $\widehat{Z}$ using network information only and not incorporating node variables, the mean square error equals to $\mathbb{E}\|\widehat{Z}-Z\|_F^2 = T_A + e_A$. Taking the difference between $\mathbb{E}\|\widehat{Z}-Z\|_F^2$ when $\widetilde{\lambda} > 0$ and $\widetilde{\lambda} = 0$, we can obtain the improvements in terms of $\mathbb{E}\|\widehat{Z}-Z\|_F^2$ by incorporating node variables:

$$
\mathbb{E}\|\widehat{Z}-Z\|_F^2 = -\widetilde{\lambda}^2(T_Y + T_A - 2T_{AY} + e_Y + e_A) + 2\widetilde{\lambda}(T_A - T_{AY} + e_A).
$$

Therefore, a smaller $(T_Y + T_A - 2T_{AY} + e_Y + e_A)$ or larger $(T_A - T_{AY} + e_A)$ woud make the improvement more siginificant. Based on the expression of each term, we can see that a larger q or a smaller C_A would lead to such cases. In other words, when the node variables contain rich enough information or the network is sparse, incorporating node variables would be more effective in terms of estimating latent variables Z.

The calculation in this Appendix is based on the assumption that we are given with fixed initial estimates $\widetilde{Z}$, $\widetilde{\alpha}$, and $\widetilde{B}$ which satisfy the conditions $\|\widetilde{Z}-Z\|_F^2 = \mathcal{O}(1)$, $\|\widetilde{\alpha}1_n^T - \alpha 1_n^T\|_F^2 = \mathcal{O}(n)$, and $\|\widetilde{B} - B\|_F^2 = \mathcal{O}(1)$. In practice, we can obtain such $\widetilde{Z}$ and $\widetilde{\alpha}$ using Algorithm 1 proposed in *Ma and Ma* (2017). $\widetilde{B}$ can be obtained by regressing Y on $\widetilde{B}$. In Appendix B.2.3 we will show the estimates obtained in such way satisfy the required condition.

B.2.2 Proof of Theorem III.9

Note that Algorithm 1 in *Ma and Ma* (2017) solves α and Z that minimizes the loss about the network part as defined in (3.6). Therefore, for the $\widetilde{Z}$ obtained from this algorithm, the graidient of L_A in (3.6) w.r.t. Z when $Z = \widetilde{Z}$ will vanish and the update of $\widetilde{Z}$ in (B.3) becomes:

$$\widehat{Z} = \widetilde{Z} + \widetilde{\lambda}\rho_2 q^{-1}\left(Y - \widetilde{Z}\widetilde{B}\right)\widetilde{B}^T. \tag{B.9}$$

We consider an ideal case that we are given a fixed $\widetilde{B}$ such that $\|\widetilde{B} - B\|_F^2 = \mathcal{O}(1)$, then the mean square error of $\widehat{Z}$ obtained in (B.9) is

$$\mathbb{E}\|\widehat{Z}-Z\|_F^2 = \mathbb{E}\left\|(I - \widetilde{\lambda}\rho_2 q^{-1}\widetilde{B}\widetilde{B}^T)(\widetilde{Z} - Z) + \widetilde{\lambda}\rho_2 q^{-1} Z(B - \widetilde{B})\widetilde{B}^T\right\|_F^2 + \widetilde{\lambda}^2\rho_2^2 nq^{-2}\sigma^2 tr(\widetilde{B}\widetilde{B}^T). \tag{B.10}$$

Based on the calculation in (B.6), (B.7), and (B.8), we have the RHS in (B.10) be bounded by

$$\mathbb{E}\|\widehat{Z} - Z\|_F^2 \leq \left((1 - \widetilde{\lambda}\rho_2\widetilde{\sigma}_k)\mathbb{E}\|\widetilde{Z} - Z\|_F + \widetilde{\lambda}\rho_2\sqrt{nq^{-1}\widetilde{\sigma}_1\lambda_1}\|B - \widetilde{B}\|_F\right)^2 + \widetilde{\lambda}^2\rho_2^2 nq^{-1}\sigma^2\widetilde{\sigma}_1. \tag{B.11}$$

Without the information from Y, i.e., set $\widetilde{\lambda} = 0$ in (B.9), we have $\mathbb{E}\|\widehat{Z} - Z\|_F^2 = \mathbb{E}\|\widetilde{Z} - Z\|_F^2$. To make sure the choice of $\widetilde{\lambda}$ would be helpful for improving the estimation of Z, we only

need to require:

$$\left((1-\widetilde{\lambda}\rho_2\widetilde{\sigma}_k)\mathbb{E}\|\widetilde{Z}-Z\|_F+\widetilde{\lambda}\rho_2\sqrt{nq^{-1}\widetilde{\sigma}_1\lambda_1}\|B-\widetilde{B}\|_F\right)^2+\widetilde{\lambda}^2\rho_2^2nq^{-2}\sigma^2\widetilde{\sigma}_1\leq\mathbb{E}\|\widetilde{Z}-Z\|_F^2.$$

This is equivalent to

$$\begin{aligned}((1-\widetilde{\lambda}\rho_2\widetilde{\sigma}_k)^2-1)\mathbb{E}\|\widetilde{Z}-Z\|_F^2+\widetilde{\lambda}^2\rho_2^2nq^{-1}\widetilde{\sigma}_1\lambda_1\|B-\widetilde{B}\|_F^2+\\ 2(1-\widetilde{\lambda}\rho_2\widetilde{\sigma}_k)\mathbb{E}\|\widetilde{Z}-Z\|_F\widetilde{\lambda}\rho_2\sqrt{nq^{-1}\widetilde{\sigma}_1\lambda_1}\|B-\widetilde{B}\|_F+\widetilde{\lambda}^2\rho_2^2nq^{-1}\sigma^2\widetilde{\sigma}_1\leq 0,\end{aligned}$$

or

$$\begin{aligned}\widetilde{\lambda}^2\rho_2^2\left(\left(\widetilde{\sigma}_k\mathbb{E}\|\widetilde{Z}-Z\|_F-\sqrt{nq^{-1}\widetilde{\sigma}_1\lambda_1}\|B-\widetilde{B}\|_F\right)^2+nq^{-1}\sigma^2\widetilde{\sigma}_1\right)+\\ 2\widetilde{\lambda}\rho_2\left(\mathbb{E}\|\widetilde{Z}-Z\|_F\sqrt{nq^{-1}\widetilde{\sigma}_1\lambda_1}\|B-\widetilde{B}\|_F-\widetilde{\sigma}_k\mathbb{E}\|\widetilde{Z}-Z\|_F^2\right)\leq 0.\end{aligned}\tag{B.12}$$

We denote

$$\bar{\lambda}=\frac{2\left(\widetilde{\sigma}_k\mathbb{E}\|\widetilde{Z}-Z\|_F^2-\|\widetilde{Z}-Z\|_F\sqrt{nq^{-1}\widetilde{\sigma}_1\lambda_1}\|B-\widetilde{B}\|_F\right)}{\rho_2\left(\left(\widetilde{\sigma}_k\mathbb{E}\|\widetilde{Z}-Z\|_F-\sqrt{nq^{-1}\widetilde{\sigma}_1\lambda_1}\|B-\widetilde{B}\|_F\right)^2+nq^{-1}\sigma^2\widetilde{\sigma}_1\right)},$$

then the inequality (B.12) hold for all $\widetilde{\lambda}$ between 0 and $\bar{\lambda}$, in other words, $\mathbb{E}\|\widehat{Z}-Z\|_F^2$ obtained in (B.10) is smaller than $\mathbb{E}\|\widetilde{Z}-Z\|_F^2$. In particular, we require $\bar{\lambda}>0$ as a positive $\widetilde{\lambda}$ implies the preference of incorporating node variables. Note $\bar{\lambda}>0$ is equivalent to

$$\widetilde{\sigma}_k\mathbb{E}\|\widetilde{Z}-Z\|_F^2-\mathbb{E}\|\widetilde{Z}-Z\|_F\sqrt{n\widetilde{\sigma}_1\lambda_1/q}\|B-\widetilde{B}\|_F>0,$$

which will hold when

$$q>\frac{\widetilde{\sigma}_1\lambda_1\|B-\widetilde{B}\|_F^2}{\widetilde{\sigma}_k\mathbb{E}\|\widetilde{Z}-Z\|_F^2}n.$$

Based on the properties of $\widetilde{Z}$ and $\widetilde{B}$, we have $C=\dfrac{\widetilde{\sigma}_1\lambda_1\|B-\widetilde{B}\|_F^2}{\widetilde{\sigma}_k\mathbb{E}\|\widetilde{Z}-Z\|_F^2}=\mathcal{O}(1)$. Then we can conclude that there exists a constant C such that when $q>Cn$, for a range of positive $\widetilde{\lambda}$, we have the updated mean square error of $\widehat{Z}$ be smaller than that of $\widetilde{Z}$. This implies that when the dimension of node variables is large enough, after we obtain an estimated $\widetilde{Z}$ from *Ma and Ma* (2017), a one-step further update of $\widetilde{Z}$ by incorporating node variables

information would guarantee the improvements in estimation of Z.

B.2.3 Lemmas on Initial Estimates with Good Properties

The calculation in Appendix B.2.1 and B.2.2 are based on the assumption that we are given initial estimates $\widetilde{Z}$, $\widetilde{\alpha}$ and $\widetilde{B}$ that are close enough to the true parameters, in the sense that $\|\widetilde{Z} - Z\|_F^2 = \mathcal{O}(1)$, $\|\widetilde{\alpha}1_n^T - \alpha 1_n^T\|_F^2 = \mathcal{O}(n)$, and $\|\widetilde{B} - B\|_F^2 = \mathcal{O}(1)$. In this section, we show such initial estimates can be obtained from existing algorithms proposed in the literature.

Lemma B.2. *The estimated $\widetilde{Z}_0$ from Algorithm 1 in Ma and Ma (2017) satisfies the condition that $dist(Z, \widetilde{Z}_0) = \min_{R,R^TR=RR^T=I_k} \|\widetilde{Z}_0R - Z\|_F^2 = \mathcal{O}_p(1)$.*

Proof. Theorem 4.2 in *Ma and Ma* (2017) suggests $\|ZZ^T - \widetilde{Z}_0\widetilde{Z}_0^T\|_F^2 = \mathcal{O}_p(n)$. Based on Lemma 8.9 in *Ma and Ma* (2017), we have

$$dist(Z, \widetilde{Z}_0)^2 \leq \|ZZ^T - \widetilde{Z}_0\widetilde{Z}_0^T\|_F^2 / \left(2(\sqrt{2}-1)^2\sigma_k^2(Z)\right) = \mathcal{O}_p(1),$$

since $\sigma_k^2(Z) = \mathcal{O}(n)$ by Assumption III.7. □

Denote $\widehat{R}_0 = \arg\min_{R,RR^T=R^TR=I_k} \|\widetilde{Z}_0R - Z\|_F^2$. Denote $\widetilde{Z}_1 = \widetilde{Z}_0\widehat{R}_0$. Assume there exists a rotation matrix $\widehat{R}_1$ such that the covariance matrix, i.e., $\widehat{R}_1^T\widetilde{Z}_1^T\widetilde{Z}_1\widehat{R}_1/n$ of $\widetilde{Z}_1\widehat{R}_1$, is diagonal. Denote $\widetilde{Z}_2 = \widetilde{Z}_1\widehat{R}_1$. We have the following lemma:

Lemma B.3. $\|\widetilde{Z}_2 - Z\|_F^2 = \mathcal{O}_p(1)$.

Proof. We have assumed that Z^TZ/n is a diagonal matrix. Since $\|\widetilde{Z}_1 - Z\|_F^2 = \mathcal{O}_p(1)$, then we have $\widetilde{Z}_1 = Z + \Delta Z_1$ with $\|\Delta Z_1\|_F^2 = \mathcal{O}_p(1)$. Since $\widetilde{Z}_1^T\widetilde{Z}_1/n = Z^TZ/n + 2Z^T\Delta Z_1/n + \Delta Z_1^T\Delta Z_1/n$, and $\|Z^T\Delta Z_1/n\|_F^2 \leq \|Z\|_F^2\|\Delta Z_1\|_F^2/n^2 = \mathcal{O}_p(1/n)$. So $\|\widetilde{Z}_1^T\widetilde{Z}_1/n - Z^TZ/n\|_F^2 = \mathcal{O}_p(1/n)$. The eigenvectors of Z^TZ/n is I, denote the eigenvalue decomposition of $\widetilde{Z}_1^T\widetilde{Z}_1/n$ is $U\Lambda U^T$, then based on Lemma B.5, we have $\|I - U\|_F^2 = \mathcal{O}_p(1/n)$. We could right multiply $\widetilde{Z}_1$ by U such that $U^T\widetilde{Z}_1^T\widetilde{Z}_1U$ is diagonal. Denote $\widetilde{Z}_2 = \widetilde{Z}_1U$. Then $\|Z - \widetilde{Z}_2\| = \|Z - \widetilde{Z}_1U\|_F^2 = \|Z - \widetilde{Z}_1 + \widetilde{Z}_1 - \widetilde{Z}_1U\|_F^2 \leq \|Z - \widetilde{Z}_1\|_F^2 + \|\widetilde{Z}_1\|_F^2\|I - U\|_F^2 = \mathcal{O}_p(1) + \mathcal{O}_p(n)\mathcal{O}_p(1/n) = \mathcal{O}_p(1)$. □

Lemma B.2 and B.3 indicate that for an $\widetilde{Z}$ estimated from Algorithm 1 in *Ma and Ma* (2017), by appropriate rotation it can satisfy both the identifiability condition in our proposed model and the condition that $\|\widetilde{Z} - Z\|_F^2 = \mathcal{O}_p(1)$.

Lemma B.4. *The estimated $\widetilde{\alpha}$ from Algorithm 1 in Ma and Ma (2017) satisfy the condition* $\|\widetilde{\alpha}1_n^T - \alpha 1_n^T\|_F^2 = \mathcal{O}_p(n)$

Proof. This is a direct result from the results of Theorem 4.2 in *Ma and Ma* (2017). □

Lemma B.5. *Consider a matrix $M \in \mathbb{R}^{k\times k} = \mathcal{O}(1)$ with u_{0j} being its jth eigenvector. And $M = M_0 + \Delta M$ with $\|\Delta M\|_F^2 = \mathcal{O}_p(1/n)$ and u_j being its jth eigenvector. Then $\|u_j - u_{0j}\|_2 = \mathcal{O}_p(1/\sqrt{n})$, for $j = 1, \ldots, k$.*

Proof. Consider a matrix $M_0 \in \mathbb{R}^{k\times k}$. λ_0 is its jth eigenvalue, and x_0 is the corresponding eigenvectors, i.e., $M_0 x_0 = \lambda_0 x_0$. Assume $M = M_0 + \Delta M$ and $\|\Delta M\|_F^2 = \mathcal{O}_p(1/n)$. λ is jth eigenvalue of M and x is the corresponding eigenvector, i.e., $Mx = \lambda x$. By Weyl's inequality, we have $|\Delta\lambda| = |\lambda - \lambda_0| = \mathcal{O}_p(1/\sqrt{n})$. Denote $x = x_0 + \Delta x$. We have

$$
\begin{aligned}
Mx &= \Lambda x \\
(M_0 + \Delta M)(x_0 + \Delta x) &= (\lambda_0 + \Delta\lambda)(x_0 + \Delta x) \\
M_0 x_0 + \Delta M x_0 + M_0 \Delta x + \Delta M \Delta x &= \lambda_0 x_0 + \Delta\lambda x_0 + \lambda_0 \Delta x + \Delta\lambda\Delta x \\
M_0\Delta x - \lambda_0 \Delta x &= \Delta\lambda x_0 + \Delta\lambda\Delta x - \Delta M x_0 - \Delta M \Delta x
\end{aligned}
$$

For the RHS, $\|\Delta\lambda x_0 + \Delta\lambda\Delta x - \Delta M x_0 - \Delta M\Delta x\|_2 \le \mathcal{O}_p(1/\sqrt{n}) + \|\Delta x\|\mathcal{O}_p(1/\sqrt{n})$. For the left hand side, $\|M_0\Delta x - \lambda_0\Delta x\| = \mathcal{O}_p(\|\Delta x\|_2)$. Therefore, $\|\Delta x\|_2 = \mathcal{O}_p(1/\sqrt{n})$. □

Lemma B.6. *Assume $Y = ZB + E$ as specified in (3.2). Assume we have $\widetilde{Z}$ such that $\|\widetilde{Z} - Z\|_F^2 = \mathcal{O}_p(1)$. If we regress Y on $\widetilde{Z}$ to obtain $\widetilde{B}$, then we have $\|\widetilde{B} - B\|_F^2 = \mathcal{O}_p(1)$.*

Proof. Since $\widetilde{B}_j = \arg\min_{B_j\in\mathbb{R}^k} \|Y_j - \widetilde{Z}B_j\|_F^2$, so $\widetilde{B}_j$ satisfies the condition $\widetilde{Z}^T(Y_j - \widetilde{Z}\widetilde{B}_j) = 0$. This is equivalent to

$$\widetilde{Z}^T(Y_j - ZB_j + ZB_j - \widetilde{Z}\widetilde{B}_j) = 0$$

Note $Y_j - Z^T B_j = E_j = \mathcal{O}_p(1)$ and each element is independent, then $\|\widetilde{Z}^T(Y_j - ZB_j)\|_F = \mathcal{O}_p(n)$ based on $\|\widetilde{Z} - Z\|_F^2 = \mathcal{O}_p(1)$ and Assumption III.7. Therefore, $\|\widetilde{Z}^T(ZB_j - \widetilde{Z}\widetilde{B}_j)\|_F^2$ should be of $\mathcal{O}_p(n)$ to make the equation hold. Since $ZB_j - \widetilde{Z}\widetilde{B}_j = ZB_j - \widetilde{Z}B_j + \widetilde{Z}B_j - \widetilde{Z}\widetilde{B}_j = (Z - \widetilde{Z})B_j + \widetilde{Z}(B_j - \widetilde{B}_j)$, and we have

$$\begin{aligned}\|\widetilde{Z}^T(Z - \widetilde{Z})B_j\|_F^2 &= B_j^T(Z - \widetilde{Z})^T\widetilde{Z}\widetilde{Z}^T(Z - \widetilde{Z})B_j \\ &\leq n\widetilde{\sigma}_1 B_j^T(Z - \widetilde{Z})^T(Z - \widetilde{Z})B_j \\ &\leq n\widetilde{\sigma}_1\mathcal{O}_p(1)\|B_j\|_F^2 = \mathcal{O}_p(n)\end{aligned}$$

Then by similar calculation we need $\|B_j - \widetilde{B}_j\|_F^2 = \mathcal{O}_p\left(1/n\right)$ to satisfy the condition. Summing over all q columns of B we have $\|\widetilde{B} - B\|_F = \mathcal{O}_p\left(q/n\right) = \mathcal{O}_p(1)$ based on Assumption III.6. □

Lemma B.7. *Assume Y are generated from model (3.4) and we have $\widetilde{Z}$ such that $\|\widetilde{Z} - Z\|_F^2 = \mathcal{O}_p(1)$. If we regress Y on $\widehat{Z}$ to obtain $\widetilde{B}$, then we have $\|\widetilde{B} - B\|_F^2 = \mathcal{O}_p(1)$.*

Proof. Similar to the calculation in proof of Lemma B.6, the $\widetilde{B}_j$ that minimizes the negative log-likelihood of (3.4) satisfies the condition that $\widetilde{Z}^T(Y_j - \sigma(\widetilde{Z}\widetilde{B}_j)) = 0$. This is equivalent to

$$\widetilde{Z}^T(Y_j - \sigma(ZB_j) + \sigma(ZB_j) - \sigma(\widetilde{Z}\widetilde{B}_j)) = 0$$

$Y_j - \sigma(ZB_j) = \mathcal{O}_p(1)$, and $\sigma(ZB_j) - \sigma(\widetilde{Z}\widetilde{B}_j) = \sigma'(\xi)(ZB_j - \widehat{Z}\widehat{B}_j) \asymp (ZB_j - \widehat{Z}\widehat{B}_j)$. Then based on the same analysis in Lemma B.6, we have $\|B_j - \widetilde{B}_j\|_F^2 = \mathcal{O}_p\left(1/n\right)$ and $\|\widetilde{B} - B\|_F = \mathcal{O}_p\left(q/n\right) = \mathcal{O}_p(1)$. □

Lemma B.6 and Lemma B.7 together show that after obtaining a good estimate of Z, by regressing Y on $\widetilde{Z}$, the estimated $\widetilde{B}$ would satisfy $\|\widetilde{B} - B\|_F^2 = \mathcal{O}(1)$. These results gurantee the existence of good initial estimate of B.

APPENDIX C

Appendix of Chapter 4

C.1 Proof of Theorem IV.3

For any parameter $\mathcal{T} \in \mathcal{F}$, where the definition of $\mathcal{F}$ is given in Section 3.3, the objective function is defined as

$$\begin{aligned} l(\mathcal{T}) &= -\log P(A|\mathcal{T}) \\ &= -\sum_{r=1}^{R}\sum_{i=1}^{n}\sum_{i=1}^{n}\left\{A_{ij}^{(r)}\Theta_{ij}^{(r)} + \log\left(1-\sigma(\Theta_{ij}^{(r)})\right)\right\} \\ &= -\sum_{r=1}^{R}\sum_{i=1}^{n}\sum_{i=1}^{n}\left\{A_{ij}^{(r)}\Theta_{ij}^{(r)} - b(\Theta_{ij}^{(r)})\right\}, \end{aligned} \tag{C.1}$$

where $b(x) = \log(1+\exp(x))$.

Denote $\mathcal{T}_\star \in \mathcal{F}$ be the true parameter value, and $\widehat{\mathcal{T}}$ is obtained from (4.4), then

$$l(\widehat{\mathcal{T}}) - l(\mathcal{T}_\star) \leq 0. \tag{C.2}$$

Further, we have

$$
\begin{aligned}
& l(\mathcal{T}_\star) - l(\widehat{\mathcal{T}}) \\
=& \sum_{r=1}^{R}\sum_{i=1}^{n}\sum_{j=1}^{n}\left\{A_{ij}^{(r)}\left(\widehat{\Theta}_{ij}^{(r)} - \Theta_{\star,ij}^{(r)}\right) - \left(b(\widehat{\Theta}_{ij}^{(r)}) - b(\Theta_{\star,ij}^{(r)})\right)\right\} \\
=& \sum_{r=1}^{R}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(A_{ij}^{(r)} - b'(\Theta_{\star,ij}^{(r)})\right)\left(\widehat{\Theta}_{ij}^{(r)} - \Theta_{\star,ij}^{(r)}\right) \\
& - \sum_{r=1}^{R}\sum_{i=1}^{n}\sum_{j=1}^{n}\left\{b(\widehat{\Theta}_{ij}^{(r)}) - b(\Theta_{\star,ij}^{(r)}) - b'(\Theta_{\star,ij}^{(r)})\left(\widehat{\Theta}_{ij}^{(r)} - \Theta_{\star,ij}^{(r)}\right)\right\}.
\end{aligned}
\tag{C.3}
$$

By Taylor's expansion, the last expression in (C.3) can be expressed as

$$
\begin{aligned}
& \sum_{r=1}^{R}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(A_{ij}^{(r)} - b'(\Theta_{\star,ij}^{(r)})\right)\left(\widehat{\Theta}_{ij}^{(r)} - \Theta_{\star,ij}^{(r)}\right) - \sum_{r=1}^{R}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{1}{2}b''(\widetilde{\Theta}_{ij}^{(r)})\left(\widehat{\Theta}_{ij}^{(r)} - \Theta_{\star,ij}^{(r)}\right)^2 \\
\leq& \sum_{r=1}^{R}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(A_{ij}^{(r)} - b'(\Theta_{\star,ij}^{(r)})\right)\left(\widehat{\Theta}_{ij}^{(r)} - \Theta_{\star,ij}^{(r)}\right) - \frac{1}{2}\min_{|v|\leq\mu} b''(v)\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F^2,
\end{aligned}
\tag{C.4}
$$

where $\widetilde{\Theta}_{ij}^{(r)} = \eta_{ij}\widehat{\Theta}_{ij}^{(r)} + (1-\eta_{ij})\Theta_{\star,ij}^{(r)}$ for some $\eta_{ij} \in (0,1)$. By (C.2), (C.3) and (C.4), we have

$$
\begin{aligned}
\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F^2 &\leq \frac{2}{\min_{|v|<\mu} b''(v)}\sum_{r=1}^{R}\sum_{i=1}^{n}\sum_{i=1}^{n}\left(A_{ij}^{(r)} - b'(\Theta_{\star,ij}^{(r)})\right)\left(\widehat{\Theta}_{ij}^{(r)} - \Theta_{\star,ij}^{(r)}\right) \\
&= \frac{2}{\min_{|v|<\mu} b''(v)}\left\langle Z, \widehat{\mathcal{T}} - \mathcal{T}_\star\right\rangle.
\end{aligned}
\tag{C.5}
$$

Here we define $Z = [Z^{(1)}; Z^{(2)}; \cdots; Z^{(R)}] \in \mathbb{R}^{n\times n\times R}$ as a three-way tensor with entries $Z_{ij}^{(r)} = A_{ij}^{(r)} - b'(\Theta_{\star,ij})$, for $i, j = 1, \ldots, n, r = 1, \ldots, R$.

For notational simplicity, we decompose each $\mathcal{T} \in \mathcal{F}$ into two parts: $\mathcal{T} = \mathcal{H} + \mathcal{M}$. Here, $\mathcal{H} = [\mathcal{H}^{(1)}; \mathcal{H}^{(2)}; \cdots; \mathcal{H}^{(R)}] \in \mathbb{R}^{n\times n\times R}$ with $\mathcal{H}^{(r)} = \alpha^{(r)}1_n^\top + 1_n\alpha^{(r)\top} \in \mathbb{R}^{n\times n}$ is the term related to node degree heterogeneity parameters, and

$$
\mathcal{M} = [U\Lambda^{(1)}U^\top; U\Lambda^{(2)}U^\top; \cdots; U\Lambda^{(R)}U^\top] \in \mathbb{R}^{n\times n\times R}
$$

is the term related to shared latent representations. Therefore, the quantity $\left\langle Z, \widehat{\mathcal{T}} - \mathcal{T}_\star\right\rangle$

in (C.5) can be decomposed as

$$\begin{aligned}
&\left\langle Z, \widehat{\mathcal{T}} - \mathcal{T}_\star \right\rangle \\
=&\left\langle Z, \widehat{\mathcal{H}} - \mathcal{H}_\star \right\rangle + \left\langle Z, \widehat{\mathcal{M}} - \mathcal{M}_\star \right\rangle \\
=&\sum_{r=1}^{R} \left\langle Z^{(r)}, \widehat{\alpha}^{(r)} 1_n^\top + 1_n \widehat{\alpha}^{(r)\top} - \alpha_\star^{(r)} 1_n^\top - 1_n \alpha_\star^{(r)\top} \right\rangle + \left\langle Z, \widehat{\mathcal{M}} - \mathcal{M}_\star \right\rangle.
\end{aligned} \tag{C.6}$$

We bound two summands in (C.6) respectively. For any two matrices A and B, we have $|\langle A, B\rangle| \le \|A\|_2 \|B\|_\star \le \|A\|_2 \sqrt{rank(B)} \|B\|_F$. The definition of Z implies that entries in Z are independent, mean-zero sub-gaussian random variables with $\mathbb{E}[\exp(tZ_{ij}^{(r)})] \le \exp(t^2/8)$. Therefore, we can apply Lemma C.1 (given in Section C.1.1) to $Z^{(r)}$ for any given r, and obtain that with probability at least $1-\exp(-c_1 n)$, $\|Z^{(r)}\|_2 \le C_1' \sqrt{n}$, for absolute constants c_1 and C_1'. Then with probability at least $1 - R\exp(-c_1 n)$, we have $\max_r(\|Z^{(r)}\|_2) \le C_1' \sqrt{n}$. Thus, with probability greater than $1 - R\exp(-c_1 n)$, the first term in (C.6) can be bounded as

$$\begin{aligned}
&\sum_{r=1}^{R} \left\langle Z^{(r)}, \widehat{\alpha}^{(r)} 1_n^\top + 1_n \widehat{\alpha}^{(r)\top} - \alpha_\star^{(r)} 1_n^\top - 1_n \alpha_\star^{(r)\top} \right\rangle \\
\le& 2 \sum_{r=1}^{R} \|Z^{(r)}\|_2 \|\widehat{\alpha}^{(r)} 1_n^\top + 1_n \widehat{\alpha}^{(r)\top} - \alpha_\star^{(r)} 1_n^\top - 1_n \alpha_\star^{(r)\top}\|_F \\
\le& 2 C_1' \sqrt{n} \sum_{r=1}^{R} \|\widehat{\alpha}^{(r)} 1_n^\top + 1_n \widehat{\alpha}^{(r)\top} - \alpha_\star^{(r)} 1_n^\top - 1_n \alpha_\star^{(r)\top}\|_F \\
=& 2 C_1' \sqrt{n} \sum_{r=1}^{R} \|\widehat{\mathcal{H}}^{(r)} - \mathcal{H}_\star^{(r)}\|_F \le 2 C_1' \sqrt{n} \sqrt{R} \|\widehat{\mathcal{H}} - \mathcal{H}_\star\|_F.
\end{aligned} \tag{C.7}$$

The first inequality in (C.7) is by the fact that

$$rank\left(\widehat{\alpha}^{(r)} 1_n^\top + 1_n \widehat{\alpha}^{(r)\top} - \alpha_\star^{(r)} 1_n^\top - 1_n \alpha_\star^{(r)\top}\right) \le 4.$$

Next we bound the second term in (C.6). For any two three-way tensors A and B with same dimensions $n_1 \times n_2 \times n_3$, we have $|\langle A, B\rangle| \le \|A\|_2 \|B\|_\star$. The nuclear norm of B, $\|B\|_\star$ is further bounded by $\sqrt{r_1 r_2}\|B\|_F$, where r_1 is the rank of the matrix that stacks B along its first mode into a matrix of size $n_1 \times (n_2 n_3)$, and similarly, r_2 is the rank of the matrix that stacks B along its second mode into a matrix of size $n_2 \times (n_1 n_3)$ (*Wang et al.*,

2017b; *Wang and Li*, 2018). For any tensor $\mathcal{M} = [U\Lambda^{(1)}U^\top; U\Lambda^{(2)}U^\top; \cdots; U\Lambda^{(R)}U^\top] \in \mathbb{R}^{n\times n\times R}$, stacking it along its first mode, we could obtain an $n \times nR$ matrix $\mathcal{M}_1 = [U\Lambda^{(1)}U^\top\ U\Lambda^{(2)}U^\top\ \cdots\ U\Lambda^{(R)}U^\top]$. Since

$$\mathcal{M}_1 = [U\Lambda^{(1)}U^\top\ U\Lambda^{(2)}U^\top\ \cdots\ U\Lambda^{(R)}U^\top] = U[\Lambda^{(1)}U^\top\ \Lambda^{(2)}U^\top\ \cdots\ \Lambda^{(R)}U^\top],$$

so rank of $\mathcal{M}_1$ is k. Since each layer in $\mathcal{M}$ is symmetric, stacking the tensor along its second mode similarly yields a rank k matrix. This leads to $\|\widehat{\mathcal{M}} - \mathcal{M}\|_\star \le \sqrt{2k\cdot 2k}\|\widehat{\mathcal{M}} - \mathcal{M}\|_F = 2k\|\widehat{\mathcal{M}} - \mathcal{M}\|_F$. Additionally, applying Lemma C.1 to Z, we have with probability at least $1-\exp(-c_2(2n+R))$, $\|Z\|_2 \le C_2'\sqrt{2n+R}$ for absolute constants c_2 and C_2'. Therefore, the second term in (C.6) can be bounded as

$$\left\langle Z, \widehat{\mathcal{M}} - \mathcal{M}_\star \right\rangle \le \|Z\|_2\|\widehat{\mathcal{M}} - \mathcal{M}_\star\|_\star \le 2k\|Z\|_2\|\widehat{\mathcal{M}} - \mathcal{M}_\star\|_F \le 2kC_2'\sqrt{2n+R}\|\widehat{\mathcal{M}} - \mathcal{M}_\star\|_F \tag{C.8}$$

with probability at least $1-\exp(-c_2(2n+R))$.

Plugging (C.6), (C.7) and (C.8) into (C.5) yields

$$\begin{aligned}\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F^2 &\le \frac{2}{\min_{|v|<\mu} b''(v)}\left\langle Z, \widehat{\mathcal{T}} - \mathcal{T}_\star\right\rangle \\ &\le \frac{2}{\min_{|v|<\mu} b''(v)}\left(2C_1'\sqrt{nR}\|\widehat{\mathcal{H}} - \mathcal{H}_\star\|_F + 2kC_2'\sqrt{2n+R}\|\widehat{\mathcal{M}} - \mathcal{M}_\star\|_F\right).\end{aligned} \tag{C.9}$$

Lemma C.2 in Section C.1.1 implies that $\|\widehat{\mathcal{H}} - \mathcal{H}_\star\|_F \le \|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F$, so does $\|\widehat{\mathcal{M}} - \mathcal{M}_\star\|_F \le \|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F$. Dividing both sides in (C.9) by $\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F$ leads to

$$\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F \le \frac{2}{\min_{|v|<\mu} b''(v)}\left(2C_1'\sqrt{nR} + 2kC_2'\sqrt{2n+R}\right). \tag{C.10}$$

Taking the square of both sides, we can conclude that with probability at least $1 - R\exp(-c_1 n) - \exp(-c_2(2n+R))$, we have

$$\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F^2 \le C_1 nR + C_2(2n+R), \tag{C.11}$$

where $C_1 = 32(C_1')^2/(\min_{|v|<\mu} b''(v))^2$ and $C_2 = 32k^2(C_2')^2/(\min_{|v|<\mu} b''(v))^2$.

C.1.1 Lemmas for Theorem IV.3

This subsection includes lemmas that are used in the proof of Theorem IV.3.

Lemma C.1 (Theorem 1 in *Tomioka and Suzuki* (2014))**.** *Let $\mathcal{X} \in \mathbb{R}^{n_1 \times \cdots \times n_K}$ be a K-way tensor. Assume each element $\mathcal{X}_{i_1 \ldots i_k}$ is independent, zero mean and satisfies $\mathbb{E}[e^{t\mathcal{X}_{i_1 \cdots i_k}}] \leq \exp(\sigma^2 t^2/2)$. Then there exist constants c and C which only depend on σ^2 and K such that with probability at least $1 - \exp(c\sum_k n_k)$, the spectral norm of $\mathcal{X}$ is bounded by $\|\mathcal{X}\|_2 \leq C\sqrt{\sum_{k=1}^K n_k}$.*

Lemma C.2. *For $\widehat{\mathcal{T}}, \mathcal{T}_\star \in \mathcal{F}$, when decomposing $\widehat{\mathcal{T}} = \widehat{\mathcal{H}} + \widehat{\mathcal{M}}$ and $\mathcal{T}_\star = \mathcal{H}_\star + \mathcal{M}_\star$, we have the following identity:*

$$\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F^2 = \|\widehat{\mathcal{H}} - \mathcal{H}_\star\|_F^2 + \|\widehat{\mathcal{M}} - \mathcal{M}_\star\|_F^2.$$

Proof. Since for any $\mathcal{T} \in \mathcal{F}$ we require $JU = U$, then $U^\top 1_n = 0$. Then we have

$$\widehat{\mathcal{M}}^{(r)} 1_n = \mathcal{M}_\star^{(r)} 1_n = 0,$$

and

$$1_n^\top \widehat{\mathcal{M}}^{(r)} = 1_n^\top \mathcal{M}_\star^{(r)} = 0$$

for $r = 1, \ldots, R$. Therefore,

$$\|\widehat{\Theta}^{(r)} - \Theta_\star^{(r)}\|_F^2 = \|\widehat{\mathcal{H}}^{(r)} - \mathcal{H}_\star^{(r)}\|_F^2 + \|\widehat{\mathcal{M}}^{(r)} - \mathcal{M}_\star^{(r)}\|_F^2. \tag{C.12}$$

Summing (C.12) over r gives

$$\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F^2 = \|\widehat{\mathcal{H}} - \mathcal{H}_\star\|_F^2 + \|\widehat{\mathcal{M}} - \mathcal{M}_\star\|_F^2.$$

□

C.2 Proofs of Theorem IV.4 and Corollary IV.6

Theorem IV.3 and Lemma C.2 imply that with probability at least $1 - R\exp(-c_1 n) - \exp(-c_2(2n+R))$,

$$\|\widehat{\mathcal{M}} - \mathcal{M}_\star\|_F^2 \le \|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F^2 \le C_1 nR + C_2(2n+R),$$

or,

$$\frac{1}{R}\sum_{r=1}^{R} \|\widehat{\mathcal{M}}^{(r)} - \mathcal{M}_\star^{(r)}\|_F^2 \le C_1 n + (2+\delta)C_2 nR^{-1} = C_1 n + \widetilde{C}_2 nR^{-1} \tag{C.13}$$

where $\widetilde{C}_2 = C_2(2+\delta)$ by the assumption that $R \le \delta n$. Therefore, there must exist a $r_0 \in \{1, \cdots, R\}$, such that

$$\|\widehat{\mathcal{M}}^{(r_0)} - \mathcal{M}_\star^{(r_0)}\|_F^2 \le C_1 n + \widetilde{C}_2 nR^{-1}. \tag{C.14}$$

The assumptions $\sigma_{\min}(\Lambda_\star^{(r_0)}) \ge \kappa$ and $U_\star^\top U_\star = nI_k$ imply that

$$\sigma_k(\mathcal{M}_\star^{(r_0)}) = \sigma_k(U_\star \Lambda_\star^{(r_0)} U_\star^\top) \ge n\kappa. \tag{C.15}$$

We also note that

$$\sigma_{k+1}(\mathcal{M}_\star^{(r_0)}) = 0 \tag{C.16}$$

since $M_\star$ is of rank k.

Combining (C.13) to (C.16), together with Davis-Kahan Theorem (*Davis and Kahan*, 1970; *Yu et al.*, 2015), we have

$$\begin{aligned} &\min_{O:O^\top O = OO^\top = I_k} \left\{ \|\widehat{U} - U_\star O\|_F^2 \right\} \le \frac{8n\|\widehat{\mathcal{M}}^{(r_0)} - \mathcal{M}_\star^{(r_0)}\|_F^2}{\{\sigma_k(\mathcal{M}_\star^{(r_0)}) - \sigma_{k+1}(\mathcal{M}_\star^{(r_0)})\}^2} \\ \le& 8\frac{C_1 n^2 + \widetilde{C}_2 n^2 R^{-1}}{\kappa^2 n^2} = 8\kappa^{-2}(C_1 + \widetilde{C}_2 R^{-1}). \end{aligned} \tag{C.17}$$

This leads to the results in Theorem IV.4.

For Corrollary IV.6, note that when $\alpha^{(r)} = 0$ for $r = 1, \ldots, R$, we have $\mathcal{T} = \mathcal{M}$. Then all the terms related to the node degree heterogeneity parameters in the calculations in

Section C.1 would be dropped and we would obtain

$$\|\widehat{\mathcal{T}} - \mathcal{T}_\star\|_F^2 \leq C(2n + R),$$

with probability $1 - c\exp(2n + R)$ for absolute constants c and C. Applying the same procedure in the proof of Theorem IV.4, we have the results in Corrollary IV.6. Also note that in the proof of Theorem IV.3, we only utilize the assumption that $JU = U$ to prove Lemma C.2. When $\mathcal{T} = \mathcal{M}$, we no longer need Lemma C.2 to obtain (C.10) from (C.9). Therefore, the assumption $JU = U$ can be disregarded in the corollary. When fitting logistic RESCAL model in real data applications, we also do not put such constraints on the estimated parameters.

C.3 Proof of Proposition IV.1

This section shows the identifiability conditions of model (4.1), as proposed in Proposition IV.1. To prove Proposition IV.1, we need the following lemma.

Lemma C.3. *For any $\beta = (\beta_1, \cdots, \beta_n)^\top \in \mathbb{R}^n$, if $\beta 1_n^\top 1_n + 1_n \beta^\top 1_n = 0_n$, then $\beta = 0_n$.*

Proof. The condition can be wriiten as

$$n \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_n \end{bmatrix} + \begin{bmatrix} \sum_{i=1}^n \beta_i \\ \vdots \\ \sum_{i=1}^n \beta_i \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \tag{C.18}$$

which implies $\beta_1 = \cdots = \beta_n = -\frac{1}{n}\sum_{i=1}^n \beta_i$. Thus we must have $\beta = 0_n$. □

By Assumption A1, we have $JU = U$, $JU_\dagger = U_\dagger$, where $J = I_n - 1_n 1_n^\top / n$. Therefore, $U\Lambda^{(r)}U^\top 1_n = U_\dagger \Lambda_\dagger^{(r)} U_\dagger^\top 1_n = 0_n$ for $r = 1, \cdots, R$. Suppose two sets of parameters yield the same edge connection probabilities, i.e.,

$$\alpha^{(r)} 1_n^\top + 1_n \alpha^{(r)\top} + U\Lambda^{(r)}U^\top = \alpha_\dagger^{(r)} 1_n^\top + 1_n \alpha_\dagger^{(r)\top} + U_\dagger \Lambda_\dagger^{(r)} U_\dagger^\top \tag{C.19}$$

for $r = 1, \ldots, R$. Right multiplying 1_n to both sides in (C.19) gives

$$\alpha^{(r)} 1_n^\top 1_n + 1_n \alpha^{(r)\top} 1_n = \alpha_\dagger^{(r)} 1_n^\top 1_n + 1_n \alpha_\dagger^{(r)\top} 1_n, \tag{C.20}$$

or

$$(\alpha^{(r)} - \alpha_\dagger^{(r)}) 1_n^\top 1_n + 1_n (\alpha^{(r)} - \alpha_\dagger^{(r)})^\top 1_n = \mathbf{0}_n. \tag{C.21}$$

Applying Lemma C.3, we have

$$\alpha_\dagger^{(r)} = \alpha^{(r)}, \ r = 1, \ldots, R. \tag{C.22}$$

(C.19) and (C.22) together imply that

$$U_\dagger \Lambda_\dagger^{(r)} U_\dagger^\top = U \Lambda^{(r)} U^\top,$$

for $r = 1, \ldots, R$. This can be further written as

$$\begin{bmatrix} U_\dagger \Lambda_\dagger^{(1)} \\ \vdots \\ U_\dagger \Lambda_\dagger^{(R)} \end{bmatrix} U_\dagger^\top = \begin{bmatrix} U \Lambda^{(1)} \\ \vdots \\ U \Lambda^{(R)} \end{bmatrix} U^\top. \tag{C.23}$$

Note that $U^\top U = n I_k$. Left multiplying $U^\top$ to the both sides of (C.23) gives

$$\begin{bmatrix} U^\top U_\dagger \Lambda_\dagger^{(1)} \\ \vdots \\ U^\top U_\dagger \Lambda_\dagger^{(R)} \end{bmatrix} U_\dagger^\top = n \begin{bmatrix} \Lambda^{(1)} \\ \vdots \\ \Lambda^{(R)} \end{bmatrix} U^\top. \tag{C.24}$$

Further multiplying both sides in (C.24) by $[\Lambda^{(1)} \ \Lambda^{(2)}] \ \cdots \ \Lambda^{(R)}] \in \mathbb{R}^{k \times (kR)}$, we have

$$\left\{ \sum_{r=1}^R (U \Lambda^{(r)})^\top U_\dagger \Lambda_\dagger^{(r)} \right\} U_\dagger^\top = n \left\{ \sum_{r=1}^R (\Lambda^{(r)})^2 \right\} U^\top. \tag{C.25}$$

$\left\{ \sum_{r=1}^R (\Lambda^{(r)})^2 \right\}$ is a positive semi-definite matrix in $\mathbb{R}^{k \times k}$. Assumption A3 implies that

$\left\{\sum_{r=1}^{R}(\Lambda^{(r)})^2\right\}$ is of full rank, and thus invertible. Therefore, (C.25) is equivalent to

$$\frac{1}{n}\left\{\sum_{r=1}^{R}(\Lambda^{(r)})^2\right\}^{-1}\left\{\sum_{r=1}^{R}(U\Lambda^{(r)})^\top U_\dagger\Lambda_\dagger^{(r)}\right\}U_\dagger^\top = U^\top. \tag{C.26}$$

Let $O = \frac{1}{n}\left\{\sum_{r=1}^{R}(\Lambda^{(r)})^2\right\}^{-1}\left\{\sum_{r=1}^{R}(U\Lambda^{(r)})^\top U_\dagger\Lambda_\dagger^{(r)}\right\} \in \mathbb{R}^{k\times k}$, then (C.26) becomes

$$OU_\dagger^\top = U^\top. \tag{C.27}$$

Furthermore, (C.27) implies that

$$OU_\dagger^\top U_\dagger O^\top = U^\top U,\ O(nI_k)O^\top = nI_k.$$

Thus we conclude $U_\dagger = UO$ for some O such that $OO^\top = O^\top O = I_k$.

Lastly, for $r = 1, \cdots, R$, we have

$$U_\dagger\Lambda_\dagger^{(r)}U_\dagger^\top = U\Lambda^{(r)}U^\top = U_\dagger O^\top\Lambda^{(r)}OU_\dagger^\top,$$

so

$$U_\dagger^\top U_\dagger\Lambda_\dagger^{(r)}U_\dagger^\top U_\dagger = U_\dagger^\top U_\dagger O^\top\Lambda^{(r)}OU_\dagger^\top U_\dagger,\ \text{or}\ (nI_k)\Lambda_\dagger^{(r)}(nI_k) = (nI_k)O^\top\Lambda^{(r)}O(nI_k),$$

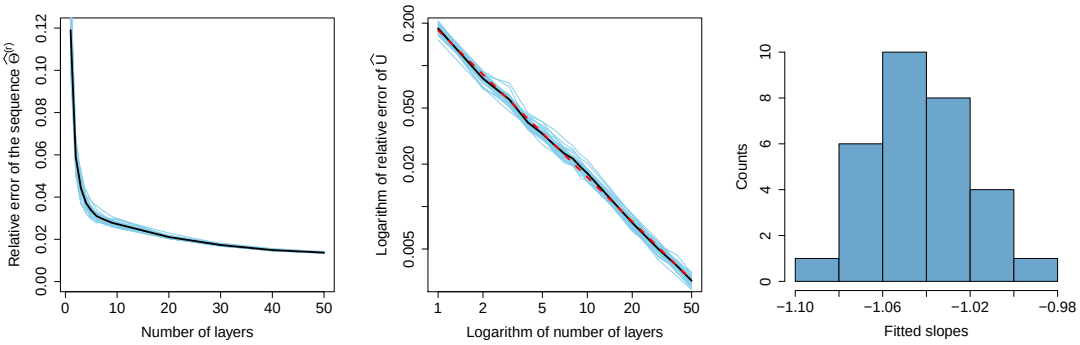
which concludes

$$\Lambda_\dagger^{(r)} = O^\top\Lambda^{(r)}O.$$

C.4 Additional Simulation Results

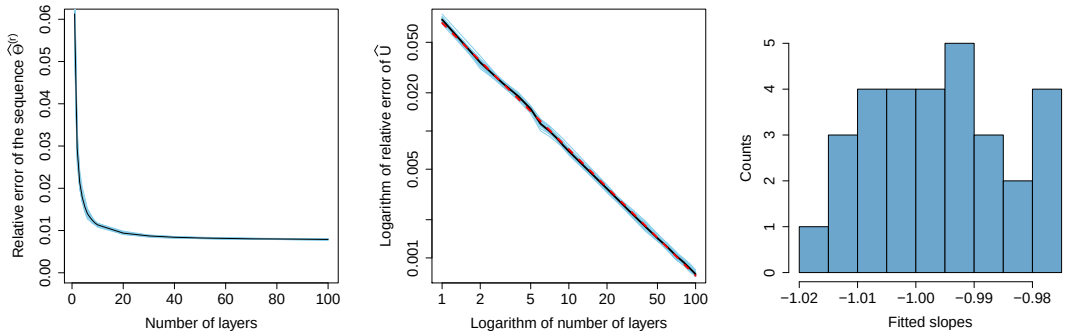
In this section we provide simulation results under the setting that $n = 200$, $R = 50$, $k = 2$ and $n = 400$, $R = 100$, $k = 4$. Figure C.1 and Figure C.2 show similar patterns of parameter estimation as we have discussed in the main article. For the estimation of the

overall connection probabilities $\{\widehat{\Theta}^{(r)}\}_{r=1}^{R_0}$, it is bounded below mainly due to the irreducible estimation error induced by the layer-specific parameters $\alpha^{(r)}$s. As for the estimation of shared latent variables U, after taking the log-log transformation of both relative error of $\widehat{U}$ and R_0, the curves can be fitted well by lines with slopes close to -1. This again demonstrates that the upper bound given in Theorem $IV.4$ is dominated by the term $\widetilde{C}_2 R^{-1}$, and the estimation error of U is inversely proportional to the number of layers used for estimation.



(a) Estimation error of $\{\widehat{\Theta}^{(r)}\}_{r=1}^{R_0}$ (b) Estimation error of $\widehat{U}$ (c) Histogram of fitted slopes

Figure C.1: (a) and (b): Estimation error of parameters when $n = 200$, $R = 50$ and $k = 2$. Each light blue curve corresponds to one replication; the black curve corresponds to the average of all replications. The red dashed line corresponds to the line whose intercept and slope equal to the average fitted intercepts and slopes. (c): Histogram of all fitted slopes.



(a) Estimation error of $\{\widehat{\Theta}^{(r)}\}_{r=1}^{R_0}$ (b) Estimation error of $\widehat{U}$ (c) Histogram of fitted slopes

Figure C.2: (a) and (b): Estimation error of parameters when $n = 400$, $R = 100$ and $k = 4$. Each light blue curve corresponds to one replication; the black curve corresponds to the average of all replications. The red dashed line corresponds to the line whose intercept and slope equal to the average fitted intercepts and slopes. (c): Histogram of all fitted slopes.

BIBLIOGRAPHY

BIBLIOGRAPHY

Airoldi, E. M., D. M. Blei, S. E. Fienberg, and E. P. Xing (2008), Mixed membership stochastic blockmodels, *Journal of machine learning research*, *9*(Sep), 1981–2014.

Allen, G. I., and R. Tibshirani (2010), Transposable regularized covariance models with an application to missing data imputation, *The Annals of Applied Statistics*, *4*(2), 764.

Allen, G. I., and R. Tibshirani (2012), Inference with transposable data: modelling the effects of row and column correlations, *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *74*(4), 721–743.

Arroyo, J., A. Athreya, J. Cape, G. Chen, C. E. Priebe, and J. T. Vogelstein (2019), Inference for multiple heterogeneous networks with a common invariant subspace, *arXiv preprint arXiv:1906.10026*.

Asur, S., and B. A. Huberman (2010), Predicting the future with social media, in *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on*, vol. 1, pp. 492–499, IEEE.

Athreya, A., D. E. Fishkind, M. Tang, C. E. Priebe, Y. Park, J. T. Vogelstein, K. Levin, V. Lyzinski, and Y. Qin (2017), Statistical inference on random dot product graphs: a survey, *Journal of Machine Learning Research*, *18*(1), 8393–8484.

Bai, J., and K. Li (2012), Statistical analysis of factor models of high dimension, *The Annals of Statistics*, *40*(1), 436–465.

Banerjee, A., A. G. Chandrasekhar, E. Duflo, and M. O. Jackson (2013), The Diffusion of Microfinance, doi:10.7910/DVN/U3BIHX.

Binkiewicz, N., J. T. Vogelstein, and K. Rohe (2017), Covariate-assisted spectral clustering, *Biometrika*, *104*(2), 361–377.

Blei, D. M., A. Kucukelbir, and J. D. McAuliffe (2017), Variational inference: A review for statisticians, *Journal of the American Statistical Association*, *112*(518), 859–877.

Bramoullé, Y., H. Djebbari, and B. Fortin (2009), Identification of peer effects through social networks, *Journal of econometrics*, *150*(1), 41–55.

Chen, J., J. Zhang, X. Xu, C. Fu, D. Zhang, Q. Zhang, and Q. Xuan (2019a), E-lstm-d: A deep learning framework for dynamic network link prediction, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.

Chen, Y., X. Li, and S. Zhang (2019b), Structured latent factor analysis for large-scale data: Identifiability, estimability, and their implications, *Journal of the American Statistical Association*, (just-accepted), 1–32.

Cheng, J., T. Li, E. Levina, and J. Zhu (2017), High-dimensional mixed graphical models, *Journal of Computational and Graphical Statistics*, *26*(2), 367–378.

Christakis, N. A., and J. H. Fowler (2007), The spread of obesity in a large social network over 32 years, *New England Journal of Medicine*, *2007*(357), 370–379.

D'Angelo, S., M. Alfò, and T. B. Murphy (2018), Node-specific effects in latent space modelling of multidimensional networks, *arXiv preprint arXiv:1807.03874*.

Davis, C., and W. M. Kahan (1970), The rotation of eigenvectors by a perturbation. iii, *SIAM Journal on Numerical Analysis*, *7*(1), 1–46.

De Bacco, C., E. A. Power, D. B. Larremore, and C. Moore (2017), Community detection, link prediction, and layer interdependence in multilayer networks, *Physical Review E*, *95*(4), 042,317.

De Lathauwer, L., B. De Moor, and J. Vandewalle (2000), A multilinear singular value decomposition, *SIAM journal on Matrix Analysis and Applications*, *21*(4), 1253–1278.

Doppa, J. R., J. Yu, P. Tadepalli, and L. Getoor (2009), Chance-constrained programs for link prediction, in *NIPS workshop on analyzing networks and learning with graphs*.

Dunn, P. K., and G. K. Smyth (2018), *Generalized Linear Models with Examples in R*, Springer.

DAngelo, S., T. B. Murphy, M. Alfò, et al. (2019), Latent space modelling of multidimensional networks with application to the exchange of votes in eurovision song contest, *The Annals of Applied Statistics*, *13*(2), 900–930.

Efron, B. (2009), Are a set of microarrays independent of each other?, *The Annals of Applied Statistics*, *3*(3), 922.

Fellinghauer, B., P. Bühlmann, M. Ryffel, M. Von Rhein, and J. D. Reinhardt (2013), Stable graphical model estimation with random forests for discrete, continuous, and mixed variables, *Computational Statistics & Data Analysis*, *64*, 132–152.

Fowler, J. H., and N. A. Christakis (2008), Dynamic spread of happiness in a large social network: longitudinal analysis over 20 years in the framingham heart study, *British Medical Journal*, *337*, a2338.

Friedman, J., T. Hastie, and R. Tibshirani (2001), *The elements of statistical learning*, vol. 1, Springer series in statistics New York.

Friedman, J., T. Hastie, and R. Tibshirani (2008), Sparse inverse covariance estimation with the graphical lasso, *Biostatistics*, *9*(3), 432–441.

Friel, N., R. Rastelli, J. Wyse, and A. E. Raftery (2016), Interlocking directorates in irish companies using a latent space model for bipartite networks, *Proceedings of the National Academy of Sciences*, *113*(24), 6629–6634.

Ghadermarzy, N., Y. Plan, and O. Yilmaz (2018), Learning tensors from partial binary measurements, *IEEE Transactions on Signal Processing*, *67*(1), 29–40.

Goldenberg, A., A. X. Zheng, S. E. Fienberg, and E. M. Airoldi (2010), A survey of statistical network models, *Foundations and Trends in Machine Learning*, *2*(2), 129–233.

Gollini, I., and T. B. Murphy (2016), Joint modeling of multiple network views, *Journal of Computational and Graphical Statistics*, *25*(1), 246–265.

Gupta, S., G. Sharma, and A. Dukkipati (2018), Evolving latent space model for dynamic networks, *arXiv:1802.03725*.

Hair, J., W. Black, B. Babin, and R. Anderson (2018), *Multivariate Data Analysis*, Cengage Learning EMEA.

Han, Q., K. Xu, and E. Airoldi (2015), Consistent estimation of dynamic and multi-layer block models, in *International Conference on Machine Learning*, pp. 1511–1520.

Handcock, M. S., A. E. Raftery, and J. M. Tantrum (2007), Model-based clustering for social networks, *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *170*(2), 301–354.

Hoff, P. (2008), Modeling homophily and stochastic equivalence in symmetric relational data, in *Advances in Neural Information Processing Systems*, pp. 657–664.

Hoff, P. D. (2003), *Random effects models for network data*, Technical report.

Hoff, P. D. (2005), Bilinear mixed-effects models for dyadic data, *Journal of the American Statistical Association*, *100*(469), 286–295.

Hoff, P. D. (2009), Multiplicative latent factor models for description and prediction of social networks, *Computational and Mathematical Organization Theory*, *15*(4), 261.

Hoff, P. D., A. E. Raftery, and M. S. Handcock (2002), Latent space approaches to social network analysis, *Journal of the American Statistical Association*, *97*(460), 1090–1098.

Holland, P. W., K. B. Laskey, and S. Leinhardt (1983), Stochastic blockmodels: First steps, *Social networks*, *5*(2), 109–137.

Hornstein, M., R. Fan, K. Shedden, and S. Zhou (2019), Joint mean and covariance estimation with unreplicated matrix-variate data, *Journal of the American Statistical Association*, *114*(526), 682–696.

Huang, W., Y. Liu, and Y. Chen (2018), Mixed membership stochastic blockmodels for heterogeneous networks, *Bayesian Analysis*.

Huizenga, H. M., J. C. De Munck, L. J. Waldorp, and R. P. Grasman (2002), Spatiotemporal eeg/meg source analysis based on a parametric noise covariance model, *IEEE Transactions on Biomedical Engineering*, *49*(6), 533–539.

Jordan, M. I., Z. Ghahramani, T. S. Jaakkola, and L. K. Saul (1999), An introduction to variational methods for graphical models, *Machine Learning*, *37*(2), 183–233.

Kalaitzis, A., J. Lafferty, N. Lawrence, and S. Zhou (2013), The bigraphical lasso, in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, pp. 1229–1237.

Karrer, B., and M. E. Newman (2011), Stochastic blockmodels and community structure in networks, *Physical review E*, *83*(1), 016,107.

Kashima, H., T. Kato, Y. Yamanishi, M. Sugiyama, and K. Tsuda (2009), Link propagation: A fast semi-supervised learning algorithm for link prediction, in *Proceedings of the 2009 SIAM international conference on data mining*, pp. 1100–1111, SIAM.

Kim, M., and J. Leskovec (2012), Latent multi-group membership graph model, in *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pp. 947–954, Omnipress.

Kolaczyk, E. D., and G. Csárdi (2014), *Statistical Analysis of Network Data with R*, vol. 65, Springer.

Kolda, T. G., and B. W. Bader (2009), Tensor decompositions and applications, *SIAM review*, *51*(3), 455–500.

Krivitsky, P. N., M. S. Handcock, A. E. Raftery, and P. D. Hoff (2009), Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models, *Social Networks*, *31*(3), 204–213.

Latała, R. (2005), Some estimates of norms of random matrices, *Proceedings of the American Mathematical Society*, *133*(5), 1273–1282.

Lazega, E., et al. (2001), *The collegial phenomenon: The social mechanisms of cooperation among peers in a corporate law partnership*, Oxford University Press on Demand.

Lee, J. D., and T. J. Hastie (2015), Learning the structure of mixed graphical models, *Journal of Computational and Graphical Statistics*, *24*(1), 230–253.

Leicht, E. A., P. Holme, and M. E. Newman (2006), Vertex similarity in networks, *Physical Review E*, *73*(2), 026,120.

Leng, C., and C. Y. Tang (2012), Sparse matrix graphical models, *Journal of the American Statistical Association*, *107*(499), 1187–1200.

Leskovec, J., and J. J. Mcauley (2012), Learning to discover social circles in ego networks, in *Advances in Neural Information Processing Systems*, pp. 539–547.

Levin, K., A. Athreya, M. Tang, V. Lyzinski, Y. Park, and C. E. Priebe (2017), A central limit theorem for an omnibus embedding of random dot product graphs, *arXiv preprint arXiv:1705.09355*.

Li, T., E. Levina, J. Zhu, et al. (2019), Prediction models for network-linked data, *The Annals of Applied Statistics*, *13*(1), 132–164.

Liben-Nowell, D., and J. Kleinberg (2007), The link-prediction problem for social networks, *Journal of the American society for information science and technology*, *58*(7), 1019–1031.

Lü, L., and T. Zhou (2011), Link prediction in complex networks: A survey, *Physica A: statistical mechanics and its applications*, *390*(6), 1150–1170.

Ma, Z., and Z. Ma (2017), Exploration of large networks with covariates via fast and universal latent space model fitting, *arXiv preprint arXiv:1705.02372*.

Manski, C. F. (1993), Identification of endogenous social effects: The reflection problem, *The review of economic studies*, *60*(3), 531–542.

McCallum, A. K., K. Nigam, J. Rennie, and K. Seymore (2000), Automating the construction of internet portals with machine learning, *Information Retrieval*, *3*(2), 127–163.

Neville, J., and D. Jensen (2000), Iterative classification in relational data, in *Proceedings of AAAI-2000 Workshop on Learning Statistical Models from Relational Data*.

Newman, M. (2010), *Networks: An Introduction*, OUP Oxford.

Newman, M. (2014), Prediction of highly cited papers, *EPL (Europhysics Letters)*, *105*(2), 28,002.

Newman, M. E. (2006), Modularity and community structure in networks, *Proceedings of the national academy of sciences*, *103*(23), 8577–8582.

Newman, M. E., and A. Clauset (2016), Structure and inference in annotated networks, *Nature Communications*, *7*, 11,863.

Newman, M. E., and M. Girvan (2004), Finding and evaluating community structure in networks, *Physical review E*, *69*(2), 026,113.

Nickel, M., and V. Tresp (2013), Logistic tensor factorization for multi-relational data, *arXiv preprint arXiv:1306.2084*.

Nickel, M., V. Tresp, and H.-P. Kriegel (2011), A three-way model for collective learning on multi-relational data, in *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pp. 809–816, Omnipress.

Nielsen, A. M., and D. Witten (2018), The multiple random dot product graph model, *arXiv preprint arXiv:1811.12172*.

Park, S., K. Shedden, and S. Zhou (2017), Non-separable covariance models for spatio-temporal data, with applications to neural encoding analysis, *arXiv preprint arXiv:1705.05265*.

Paul, S., and Y. Chen (2015), Community detection in multi-relational data with restricted multi-layer stochastic blockmodel, *arXiv preprint arXiv:1506.02699*.

Paul, S., and Y. Chen (2020), Spectral and matrix factorization methods for consistent community detection in multi-layer networks, *Annals of Statistics, in press*.

Peng, J., P. Wang, N. Zhou, and J. Zhu (2009), Partial correlation estimation by joint sparse regression models, *Journal of the American Statistical Association*, *104*(486), 735–746.

Qin, T., and K. Rohe (2013), Regularized spectral clustering under the degree-corrected stochastic blockmodel, in *Advances in Neural Information Processing Systems*, pp. 3120–3128.

Ravikumar, P., M. J. Wainwright, G. Raskutti, and B. Yu (2011), High-dimensional covariance estimation by minimizing l1-penalized log-determinant divergence, *Electronic Journal of Statistics*, *5*, 935–980.

Rothman, A. J., P. J. Bickel, E. Levina, and J. Zhu (2008), Sparse permutation invariant covariance estimation, *Electronic Journal of Statistics*, *2*, 494–515.

Rudelson, M., and S. Zhou (2017), Errors-in-variables models with dependent measurements, *Electronic Journal of Statistics*, *11*(1), 1699–1797.

Salter-Townshend, M., and T. H. McCormick (2017), Latent space models for multiview network data, *The Annals of Applied Statistics*, *11*(3), 1217.

Schönemann, P. H. (1966), A generalized solution of the orthogonal procrustes problem, *Psychometrika*, *31*(1), 1–10.

Sen, P., G. Namata, M. Bilgic, L. Getoor, B. Galligher, and T. Eliassi-Rad (2008), Collective classification in network data, *AI magazine*, *29*(3), 93.

Sewell, D. K., and Y. Chen (2015), Latent space models for dynamic networks, *Journal of the American Statistical Association*, *110*(512), 1646–1657.

Sewell, D. K., and Y. Chen (2016), Latent space models for dynamic networks with weighted edges, *Social Networks*, *44*, 105–116.

Sewell, D. K., and Y. Chen (2017), Latent space approaches to community detection in dynamic networks, *Bayesian Analysis*, *12*(2), 351–377.

Stegle, O., C. Lippert, J. M. Mooij, N. D. Lawrence, and K. M. Borgwardt (2011), Efficient inference in matrix-variate Gaussian models with\iid observation noise, in *Advances in Neural Information Processing Systems*, pp. 630–638.

Sun, Y., and J. Han (2013), Mining heterogeneous information networks: a structural analysis approach, *Acm Sigkdd Explorations Newsletter*, *14*(2), 20–28.

Tang, M., D. L. Sussman, C. E. Priebe, et al. (2013), Universally consistent vertex classification for latent positions graphs, *The Annals of Statistics*, *41*(3), 1406–1430.

Taskar, B., P. Abbeel, and D. Koller (2002), Discriminative probabilistic models for relational data, in *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pp. 485–492, Morgan Kaufmann Publishers Inc.

Tomioka, R., and T. Suzuki (2014), Spectral norm of random tensors, *arXiv preprint arXiv:1407.1870*.

Tucker, L. R. (1966), Some mathematical notes on three-mode factor analysis, *Psychometrika*, *31*(3), 279–311.

Valles-Catala, T., F. A. Massucci, R. Guimera, and M. Sales-Pardo (2016), Multilayer stochastic block models reveal the multilayer structure of complex networks, *Physical Review X*, *6*(1), 011,036.

Wackernagel, H. (2013), *Multivariate Geostatistics: An Introduction with Applications*, Springer Science & Business Media.

Wang, J., Q. Zhao, T. Hastie, A. B. Owen, et al. (2017a), Confounder adjustment in multiple hypothesis testing, *The Annals of Statistics*, *45*(5), 1863–1894.

Wang, M., and L. Li (2018), Learning from binary multiway data: Probabilistic tensor decomposition and its statistical optimality, *arXiv preprint arXiv:1811.05076*.

Wang, M., and Y. Song (2017), Tensor decompositions via two-mode higher-order svd (hosvd), in *Artificial Intelligence and Statistics*, pp. 614–622.

Wang, M., K. D. Duc, J. Fischer, and Y. S. Song (2017b), Operator norm inequalities between tensor unfoldings on the partition lattice, *Linear algebra and its applications*, *520*, 44–66.

Wang, S., J. Arroyo, J. T. Vogelstein, and C. E. Priebe (2017c), Joint embedding of graphs, *arXiv preprint arXiv:1703.03862*.

Ward, M. D., and P. D. Hoff (2007), Persistent patterns of international commerce, *Journal of Peace Research*, *44*(2), 157–175.

Ward, M. D., R. M. Siverson, and X. Cao (2007), Disputes, democracies, and dependencies: A reexamination of the kantian peace, *American Journal of Political Science*, *51*(3), 583–601.

Ward, M. D., K. Stovel, and A. Sacks (2011), Network analysis and political science, *Annual Review of Political Science*, *14*, 245–264.

Wolf, T., A. Schroter, D. Damian, and T. Nguyen (2009), Predicting build failures using social network analysis on developer communication, in *Proceedings of the 31st International Conference on Software Engineering*, pp. 1–11, IEEE Computer Society.

Xu, Z., Y. Ke, Y. Wang, H. Cheng, and J. Cheng (2012), A model-based approach to attributed graph clustering, in *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pp. 505–516, ACM.

Yang, J., J. McAuley, and J. Leskovec (2013), Community detection in networks with node attributes, in *2013 IEEE 13th International Conference on Data Mining*, pp. 1151–1156, IEEE.

Young, S. J., and E. R. Scheinerman (2007), Random dot product graph models for social networks, in *International Workshop on Algorithms and Models for the Web-Graph*, pp. 138–149, Springer.

Yu, L., W. H. Woodall, and K.-L. Tsui (2018), Detecting node propensity changes in the dynamic degree corrected stochastic block model, *Social Networks*, *54*, 209–227.

Yu, Y., T. Wang, and R. J. Samworth (2015), A useful variant of the davis–kahan theorem for statisticians, *Biometrika*, *102*(2), 315–323.

Zhang, J., and Y. Chen (2020), Modularity based community detection in heterogeneous networks, *Statistica Sinica, in press*.

Zhang, Y., E. Levina, and J. Zhu (2016), Community detection in networks with node features, *Electronic Journal of Statistics*, *10*(2), 3153–3178.

Zhao, Y., Y.-J. Wu, E. Levina, and J. Zhu (2017), Link prediction for partially observed networks, *Journal of Computational and Graphical Statistics*, *26*(3), 725–733.

Zhou, S. (2014), Gemini: Graph estimation with matrix variate normal instances, *The Annals of Statistics*, *42*(2), 532–562.

Zhou, S., P. Rütimann, M. Xu, and P. Bühlmann (2011), High-dimensional covariance estimation based on Gaussian graphical models, *Journal of Machine Learning Research*, *12*(Oct), 2975–3026.

Zhu, X., Z. Ghahramani, and J. D. Lafferty (2003), Semi-supervised learning using gaussian fields and harmonic functions, in *Proceedings of the 20th International conference on Machine learning (ICML-03)*, pp. 912–919.

Zitnik, M., M. Agrawal, and J. Leskovec (2018), Modeling polypharmacy side effects with graph convolutional networks, *Bioinformatics*, *34*(13), 457466.